보안서버 구축 가이드

- Ver 0.9 -

2007. 2

주 의 사 항

이 가이드의 사용에는 어떠한 제한도 없지만 다음과 같은 사항에 주의하여야 합니다.

○ 문서 내에 언급된 상표, 제품명 등에 대한 권리는 각 상표 또는 제품을 소유한 해당 기업에 있으며, 설명을 위해 특정 회사 제품명이나 화면이 표시된 경우 보안서버 구축을 위한 참고자료로써의 고유 목적 외에 어떠한 다른 목적도 없으며 그렇게 이용되어서도 안됩니다.

○ 문서 내에 기술된 예시 등은 일반 사용자, 기업 등에 있을 수 있는 고유한 환경을 고려하지 않았으므로 실제 환경에서는 그대로 적용되지 않을 수 있습니다. 그러므로 각 장에 기술된 내용을 적용할 때에는 먼저 각 사용자, 기업의 고유한 환경에 적합한지 확인할 필요가 있으며, 내용의 오류로 인해 발생하는 피해에 대하여 본 가이드의 발행기관은 책임을 지지 않습니다.

※ 이 가이드의 내용 중 오류를 발견하였거나 내용에 대한 의견이 있을 때에는 securekorea@kisa.or.kr로 해당 내용을 보내주시기 바랍니다.

가이드의 구성

이 가이드는 사용자들의 이해를 돕기 위하여 다음과 같이 구성되어 있습니다.

Ⅰ 장과 Ⅱ 장은 사용자들이 반드시 알아야 하는 기본적인 사항들입니다. 꼭 읽어보시고 각 업체의 환경에 적합한 보안서버를 선택해야 합니다.

보안서버 구축 방법을 선택하였다면, Ⅲ 장 ~ Ⅴ 장 중 상황에 맞는 내용을 참조하시면 됩니다. 각 장에 소개되는 설치 방법과 오류시 대처방법을 숙지한 후 보안서버 구축 전문업체에 연락하시면 보다 자세한 안내를 받을 수 있습니다.

Ⅵ 장은 보안서버를 구축한 후, 실제 웹페이지에서 수정해야 할 내용에 관한 가이드입니다. 웹페이지 적용 방법과 실제 사례를 포함하고 있으며, 보안서버가 적용되었는지 확인하는 방법을 알아보실 수 있습니다.

Ⅶ 장은 정보통신부에서 추진하고 있는 보안서버 구축 확대에 관한 FAQ를 정리한 것입니다. 2005년부터 현재까지 웹사이트 운영자들이 자주 질문하신 내용을 정리한 것이므로 우선 궁금하신 내용이 있는지 확인한 후 추가적인 문의는 보안서버전문협의회 홈페이지 (www.kisia.or.kr/secureserver)를 참조하거나 securekorea@kisa.or.kr로 문의하시기 바랍니다.

부록에는 멀티도메인 SSL 인증서와 SSL 가속기에 대한 설명이 포함되어 있습니다.

목 차

그림목차

Ⅰ. 보안서버(Secure Server)란

1. 보안서버의 정의

보안서버란 인터넷상에서 사용자 PC와 웹 서버 사이에 송수신되는 개인정보를 암호화하여 전송하는 서버를 의미합니다. 또한 보안서버는 해당 전자거래 업체의 실존을 증명하여 고객과 웹 서버간의 신뢰를 형성하고, 웹 브라우저와 웹 서버간에 전송되는 데이터의 암/복호화를 통하여 보안 채널을 형성합니다.

인터넷상에서 송·수신되는 개인정보의 대표적인 예로는 로그인시 ID/패스워드, 회원가입시 이름/전화번호, 인터넷 뱅킹 이용시 계좌 번호/계좌 비밀번호 등이 해당됩니다.

인터넷 상에서 암호화되지 않은 개인정보는 가로채기 등의 해킹을 통해 해커에게 쉽게 노출될 수 있으므로, 웹 서버에 보안서버 솔루션을 설치하면 해커가 중간에 데이터를 가로채도 암호화 되어 있어 개인정보가 노출되지 않습니다.

2. 보안서버 구축의 필요성

가. 정보유출 방지(sniffing 방지)

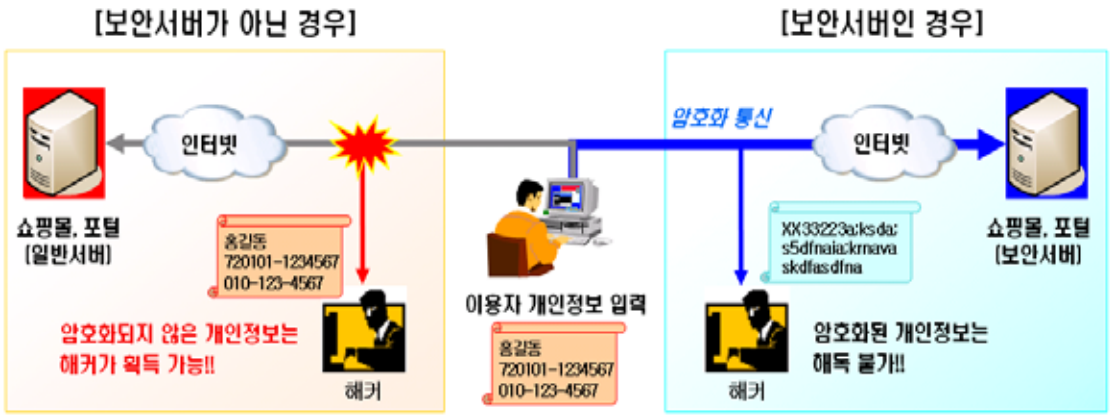
학교, PC방, 회사 등의 공용 네트워크를 사용하는 PC에서 보안서버가 구축되지 않은 사이트로 접속할 경우, 개인정보가 타인에게 노출될 가능성이 매우 높습니다. 스니핑 툴(sniffing tool)을 사용할 경우 다른 사람의 개인정보(ID/패스워드/이메일/주민번호/주소/전화번호 등)을 손쉽게 얻을 수 있습니다. 따라서 보안서버는 개인정보보호를 위해서 반드시 필요합니다.

나. 위조사이트 방지(phising 방지)

보안서버가 구축된 사이트를 이용하여 피싱(phishing) 공격을 시도하기는 어렵습니다. 따라서 보안서버가 구축된 사이트는 피싱에 의한 피해를 줄일 뿐만 아니라 고객의 신뢰를 얻을 수 있습니다.

다. 기업의 신뢰도 향상

고객의 개인정보를 안전하게 관리하는 기업이라는 이미지를 부각시킬 수 있습니다. 인터넷상의 암호화되지 않은 개인정보는 가로채기 등의 해킹을 통해 해커에게 쉽게 유출될 수 있습니다. 암호화된 개인정보는 해킹을 당해도 보호받을 수 있습니다.



<그림 1-1> 보안서버 구축의 필요성

※ 피싱(Phishing)이란, 개인정보(private data)와 낚시(fishing)를 합성한 조어로써, 금융기관 등의 웹사이트나 거기서 보내온 메일로 위장하여 개인의 ID 및 패스워드, 신용카드번호, 계좌정보 등을 빼내 이를 불법적으로 이용하는 사기수법을 뜻한다.

3. 보안서버 관련 규정

보안서버 구축 관련 규정은 아래와 같으며, 전체 법조항이 필요하신 경우는 정보통신부 홈페이지(www.mic.go.kr)나 법제처 홈페이지(www.moleg.go.kr)를 참조하시기 바랍니다.

1. 정보통신망 이용촉진 및 정보보호 등에 관한 법률

▶ 제28조(개인정보의 보호조치) 정보통신서비스제공자등은 이용자의 개인정보를 취급함에 있어서 개인정보가 분실·도난·누출·변조 또는 훼손되지 아니하도록 정보통신부령이 정하는 바에 따라 안전성 확보에 필요한 기술적·관리적 조치를 하여야 한다. <개정 2004.1.29>

▶ 제67조 (과태료) ②다음 각 호의 어느 하나에 해당하는 자는 1천만원 이하의 과태료에 처한다. <개정 2004.1.29>
8의2. 제28조의 규정을 위반하여 기술적·관리적 조치를 하지 아니한 자

2. 정보통신망 이용촉진 및 정보보호 등에 관한 법률 시행규칙

▶ 제3조의2(개인정보의 보호조치) ①법 제28조의 규정에 의한 개인정보의 안전성 확보에 필요한 기술적·관리적 조치는 다음 각호와 같다.(중간 생략)
4. 개인정보를 안전하게 저장·전송할 수 있는 암호화기술 등을 이용한 보안조치
(이하생략)

3. 개인정보의 기술적 · 관리적 보호조치 기준

▶ 제5조(개인정보의 암호화) ②정보통신서비스제공자등은 정보통신망을 통해 이용자의 개인정보 및 인증정보를 송·수신할 때에는 보안서버 구축 등의 조치를 통해 이를 암호화해야 한다. 보안서버는 다음 각호의 어느 하나의 기능을 갖추어야 한다. <개정 2007.1.29>
1. 웹서버에 SSL(Secure Socket Layer) 인증서를 설치하여 개인정보를 암호화하여 송 · 수신하는 기능
2. 웹서버에 암호화 응용프로그램을 설치하여 개인정보를 암호화하여 송 · 수신하는 기능
③정보통신서비스제공자등은 이용자의 개인정보를 PC에 저장할 때에는 이를 암호화해야 한다.

Ⅱ. 어떻게 시작하지?

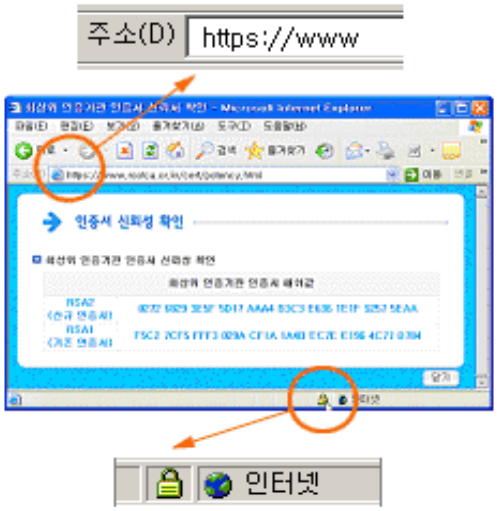
1. 보안서버의 종류

보안서버는 구축 방식에 따라 크게 「SSL 방식」과 「응용프로그램 방식」 2가지로 구분할 수 있습니다. 보안서버를 구별하는 방법은 아래와 같습니다.

가. SSL 방식

「SSL 인증서」를 이용한 보안서버는 사용자 컴퓨터에 별도 보안 프로그램 설치가 필요없으며, 웹 서버에 설치된 「SSL 인증서」를 통해 개인정보를 암호화하여 전송합니다. 보안서버 구축에 소요되는 비용이 상대적으로 저렴하지만 주기적으로 인증서 갱신을 위한 비용이 소요됩니다.

로그인 페이지 등 보안이 필요한 웹페이지에 접속한 상태에서 브라우저 하단 상태 표시줄에 자물쇠 모양의 마크로 확인할 수 있으며, 웹사이트의 구성 방법에 따라 자물쇠 모양의 마크가 보이지 않을 수 있습니다.

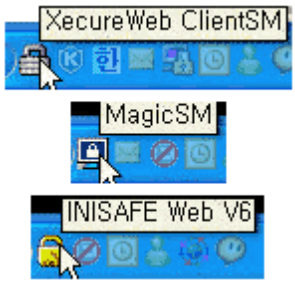


<그림 2-1> SSL 방식의 보안서버 실행 확인

나. 응용프로그램 방식

암호화 응용 프로그램을 이용한 보안서버는 웹 서버에 접속하면 사용자 컴퓨터에 자동으로 보안 프로그램이 설치되고 이를 통해 개인 정보를 암호화하여 전송합니다.

웹사이트 접속 시 초기화면이나 로그인 후 윈도우 화면 오른쪽 하단 작업표시줄 알림영역에 다음 그림과 같은 암호화 프로그램 실행여부를 확인할 수 있으며, 응용프로그램에 따라 모양은 다르게 나타날 수 있습니다.



<그림 2-2> 응용프로그램 방식의 보안서버 실행 확인

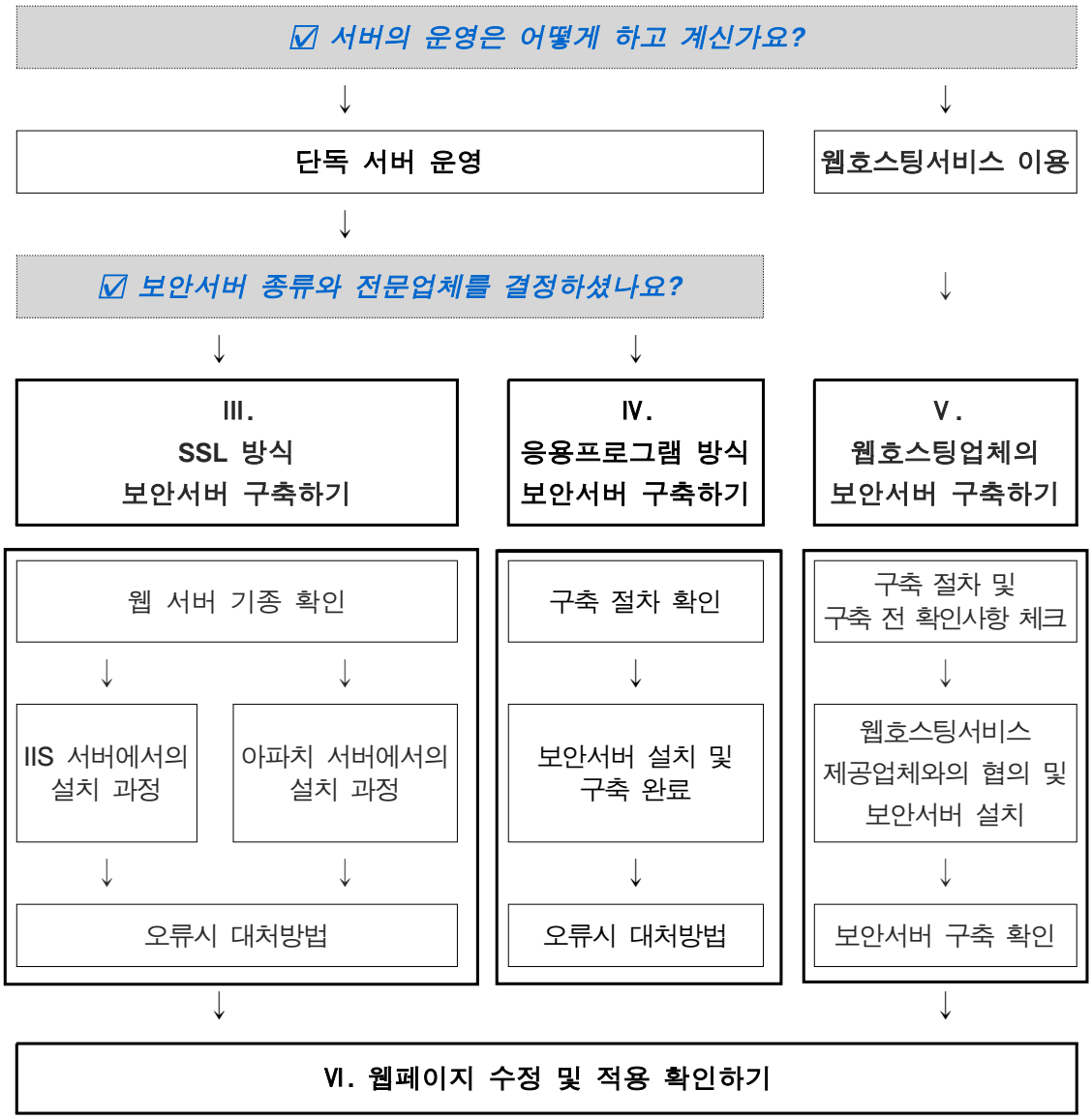
2. 보안서버 구축 전문업체

보안서버 구축 방법과 절차에 관한 보다 구체적인 내용은 다음의 「보안서버전문협의회」 회원사 중 선택하여 문의하거나 평소 알고 있는 보안서버 구축 전문업체를 통하여 자세한 설명을 받을 수 있습니다.

회사명	홈페이지	연락처
SSL 방식 솔루션 공급 업체		
한국전자인증(주)	www.crosscert.com	1588-1314
한국정보인증(주)	www.kica.net	(02) 360-3065
이모션	www.trust1.co.kr	(02) 542-1987
(주)한국무역정보통신	www.tradesign.co.kr	(02) 6000-2162
(주) 한비로	comodossl.co.kr	1544-4755
(주)닷네임코리아	www.anycert.co.kr	080-456-7770
나인포유	www.certkorea.co.kr	(02) 3444-2750
(주)아이네임즈	cert.inames.co.kr	(02) 559-1006
응용 프로그램 방식 솔루션 공급 업체		
한국전자인증(주)	www.crosscert.com	1588-1314
이니텍(주)	www.initech.com	(02) 2140-3553
한국정보인증(주)	www.signgate.com	(02) 360-3065
(주)케이사인	www.ksign.com	(02) 564-0182
드림시큐리티	www.dreamsecurity.com	(02) 2233-5533
시큐리티 테크놀로지(STI)	www.stitec.com	(02) 558-7391
펜타시큐리티시스템(주)	www.pentasecurity.com	(02) 780-7728
소프트포럼	www.softforum.co.kr	(02) 526-8423
(주)코스콤	www.signkorea.co.kr	(02) 767-7224
엠큐릭스(주)	www.mcurix.com	(02) 2253-8882
유넷시스템(주)	www.unetsystem.co.kr	(02) 390-8000

3. 보안서버 구축 절차 흐름도

지금까지 보안서버의 개념과 종류 등 보안서버를 구축하기 전에 필요한 사항들을 간단하게나마 알아보았습니다. 이제부터는 본격적으로 보안서버 구축 방법에 대하여 알아보겠습니다. 현재 기업의 상황을 확인하시고 아래 절차 흐름도를 참고하여 자신에게 필요한 내용을 찾아 각 장으로 이동하시면 됩니다.



<그림 2-3> 보안서버 구축 절차 흐름도

Ⅲ. SSL 방식 보안서버 구축하기

1. 소개 및 보안서버 구축 절차

가. 개요

SSL은 Secure Sockets Layer의 머리글이며, 1994년 Netscape에 의해 전세계적인 표준 보안 기술이 개발되었습니다.

SSL 방식은 웹 브라우저와 서버간의 통신에서 정보를 암호화함으로써 도중에 해킹을 통해 정보가 유출되더라도 정보의 내용을 보호할 수 있는 기능을 갖춘 보안 솔루션으로 전세계적으로 수 백 만개의 웹사이트에서 사용하고 있습니다.

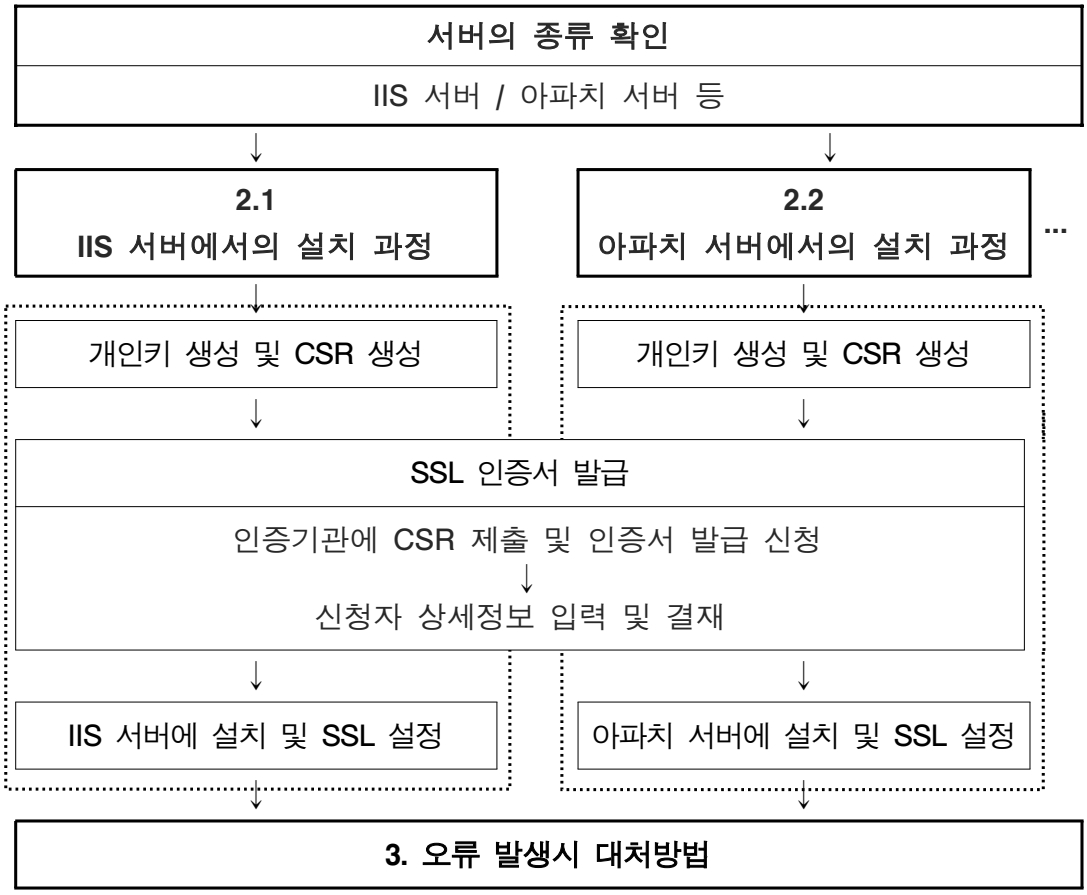
아래는 SSL 보안에 대해 그림으로 간단하게 설명해 놓은 것입니다.

<그림 3-1> SSL 방식의 보안서버 개념도

인증기관(Certification Authorities)에서 제공하는 SSL 인증서를 발급받아 웹 서버에 설치하게 되면 웹사이트 이용자들의 거래, ID/패스워드, 개인정보 등을 암호화하여 송수신할 수 있습니다.

나. 보안서버 구축 절차

SSL 방식의 보안서버 구축 절차는 다음과 같습니다.



<그림 3-2> SSL 방식 보안서버 구축 절차

① SSL 방식의 보안서버를 사용하기 위해서는 운영하고 있는 웹 서버에 보안서버 인증서가 설치되어야 합니다. 보안서버 인증서는 운영 중인 웹 서버에서 '인증서 만들기'를 이용하여 생성합니다.

※ 발급이 완료된 인증서는 재발급 또는 변경이 불가능하기 때문에 새로 발급받으셔야 하며, 새로 발급받을 시 비용이 발생할 수 있으니 CSR 생성시 절대 주의 바랍니다.

② 먼저 운영하는 웹 서버에서 개인키를 만든 후, CSR 파일을 생성하여 인증기관에 보안서버 인증서 발급을 신청합니다.

CSR(Certificate Signing Request)이란 인증서 요청파일의 약어로서 운영하는 URL 및 운영하는 회사의 정보 등이 입력됩니다.

③ 인증기관에 CSR을 이용하여 인증서를 신청할 때 회사의 담당자 정보 등을 입력합니다. 인증서 발급 심사 후에 신청 시 입력한 담당자의 E-mail 주소로 인증서가 발급됩니다.

④ 발급받은 인증서를 운영 중인 웹 서버에 설치하게 되면 SSL 방식의 보안서버 설정을 완료하게 됩니다.

서버호스팅 서비스를 받고 있는 고객의 경우에는 서버에 대한 관리자 권한이 고객에게 있기 때문에 고객이 직접 CSR 생성 및 인증서 발행 후에 설치를 진행해야 하며, 호스팅 서비스 제공업체에게 보안서버 구축 대행을 요청하게 되면 설치대행비가 부과될 수 있습니다.

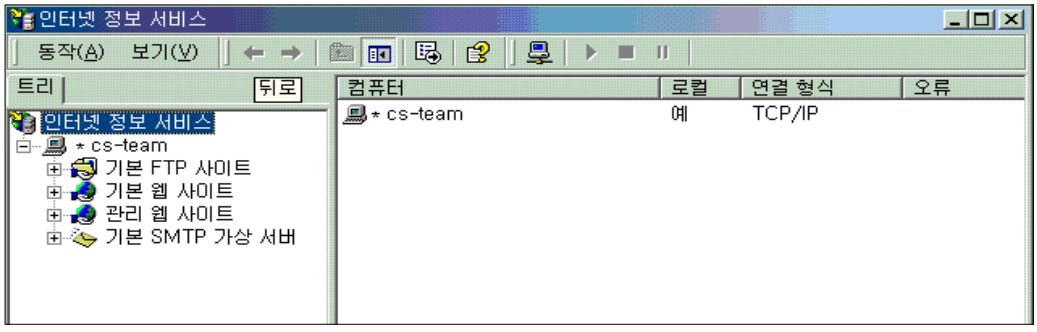
SSL 방식의 보안서버 구축은 서버의 운영체제에 따라 적용절차가 모두 다르므로, 가장 많이 사용되는 IIS와 아파치 서버를 중심으로 설명하겠습니다. IIS 서버에 설치하는 경우는 2.1의 내용을 참고하시고, 아파치 서버에 설치하는 경우는 2.2의 내용을 참고하시기 바랍니다. 향후 다른 종류의 서버에 SSL 방식의 보안서버를 설치하는 방법을 추가해 나갈 예정입니다.

2. 설치 과정

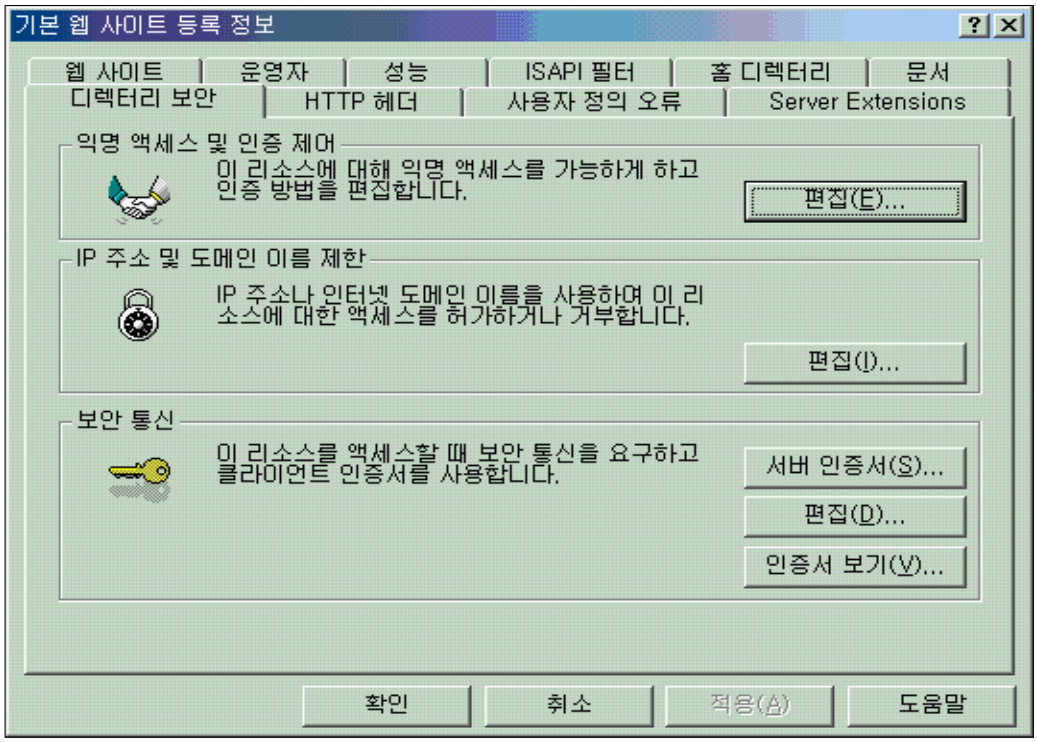
2.1 IIS 서버에서의 설치 과정

가. 개인키 생성 및 CSR 생성 방법

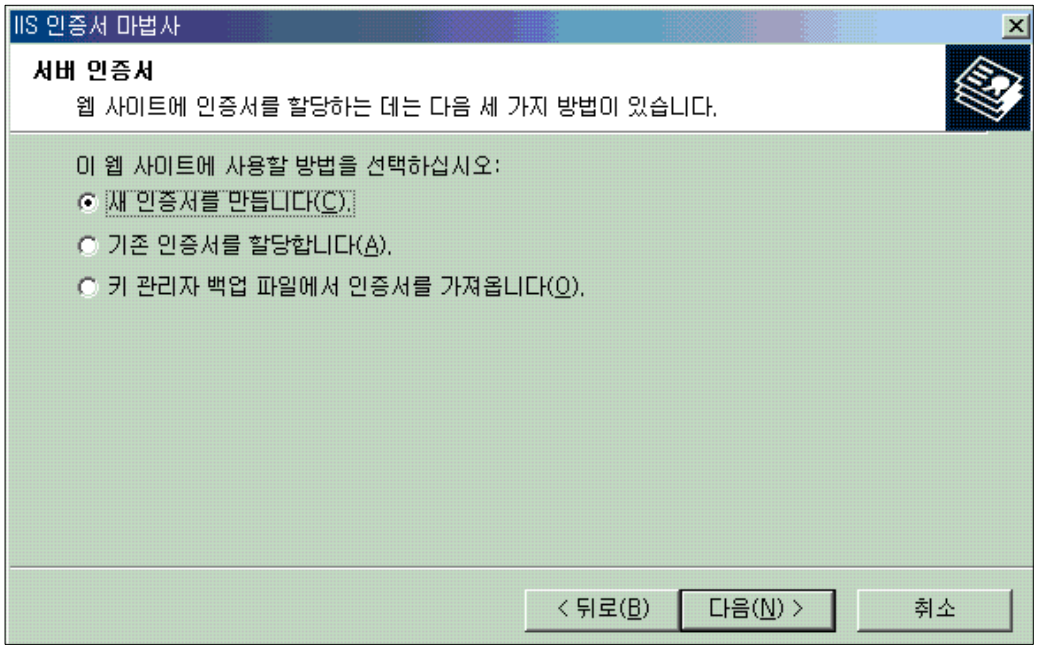
① 웹사이트 속성 선택
시작→프로그램→관리도구→인터넷 서비스 관리자→웹사이트→속성



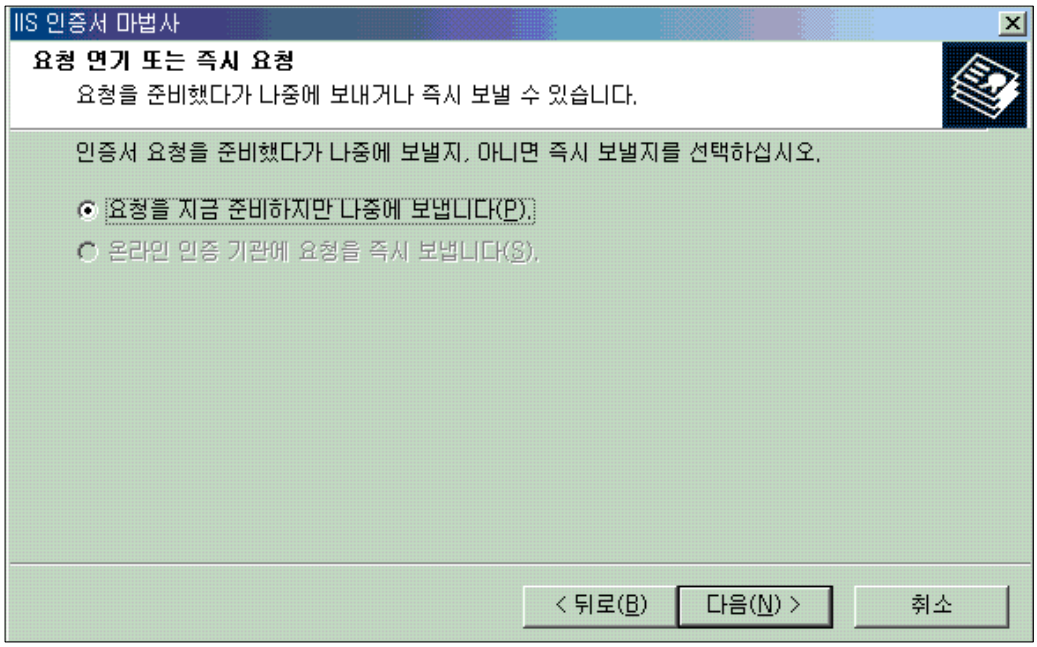
② 등록정보 화면에서 디렉토리 보안을 클릭한 후 서버 인증서를 클릭합니다.



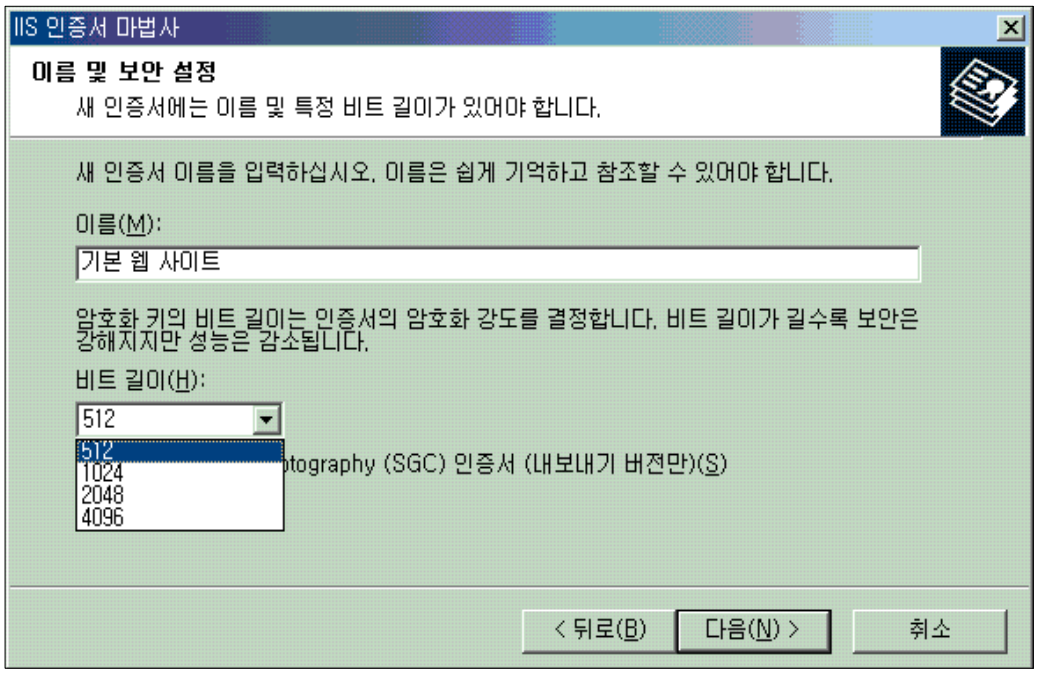
③ 웹 서버 인증서 마법사를 시작합니다. '새 인증서를 만듭니다'를 선택합니다.



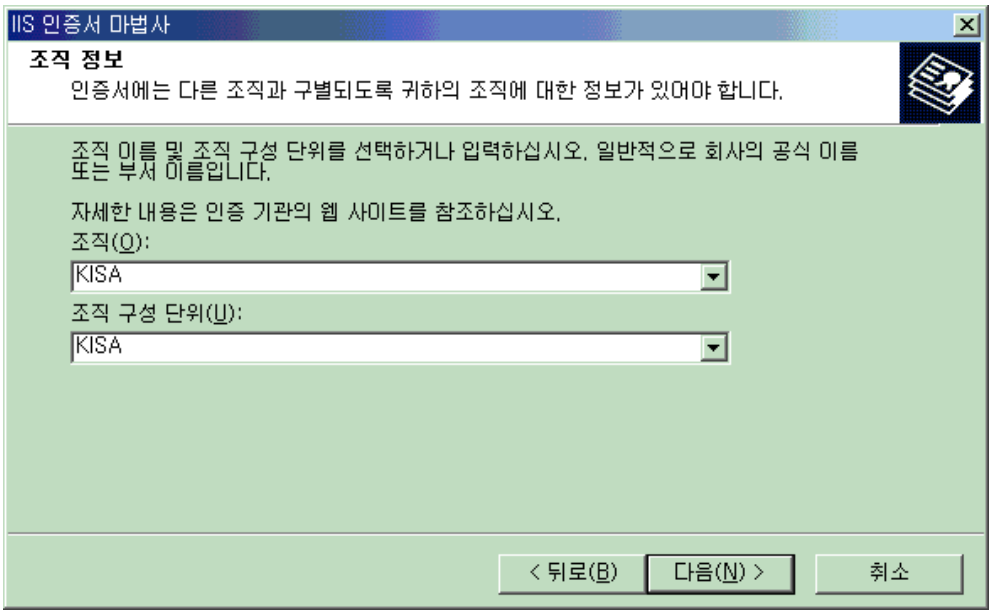
④ '요청을 준비하지만 나중에 보냅니다'를 선택합니다.



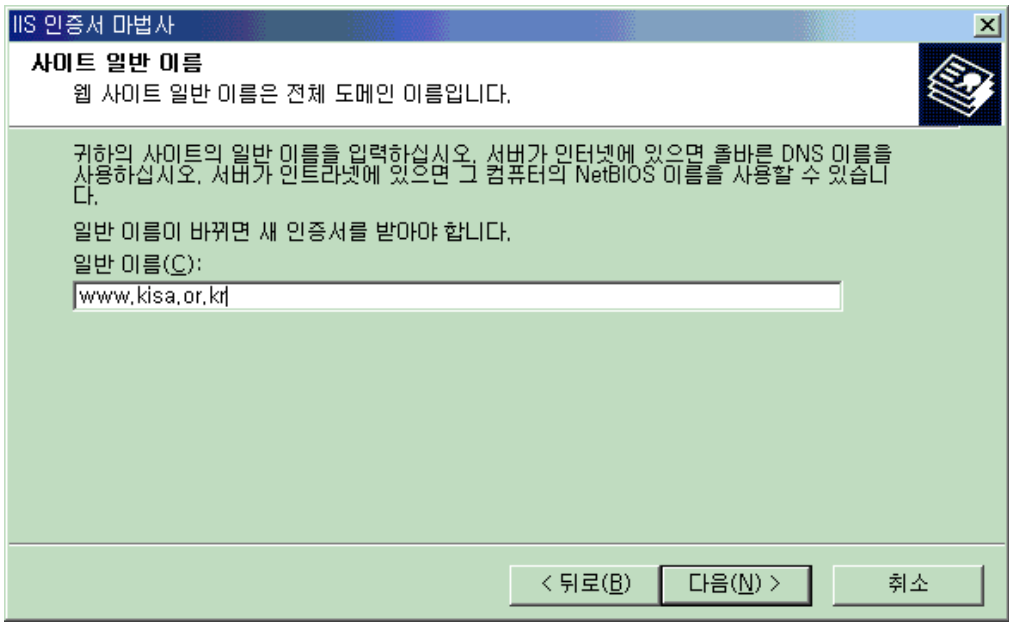
⑤ 인증서를 만들 이름을 입력하시기 바랍니다.
이름은 인증서의 별칭이므로 쉬운 것으로 입력하여 주시기 바랍니다.
인증서 키의 길이는 1,024가 표준입니다. 비트 길이가 너무 크면 서버에서 인지하지 못할 경우도 있습니다.



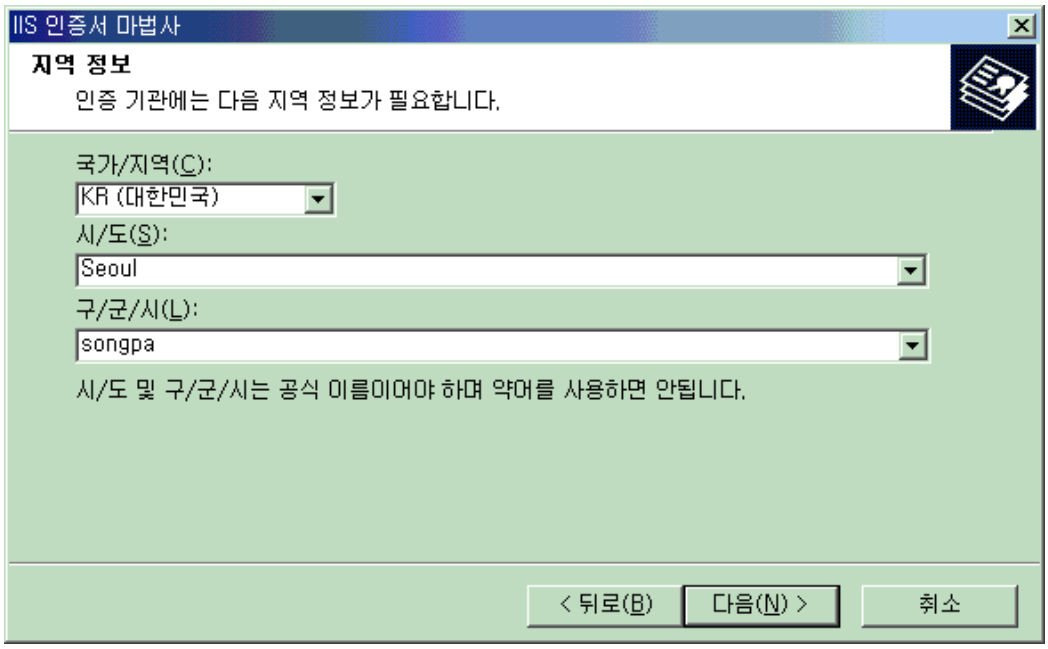
⑥ 조직 및 조직 구성 단위를 입력합니다.
조직은 회사의 영문 전체 이름을 입력하고, 조직 구성단위는 영문 부서명을 입력합니다.(모든 내용은 영문으로 입력합니다)



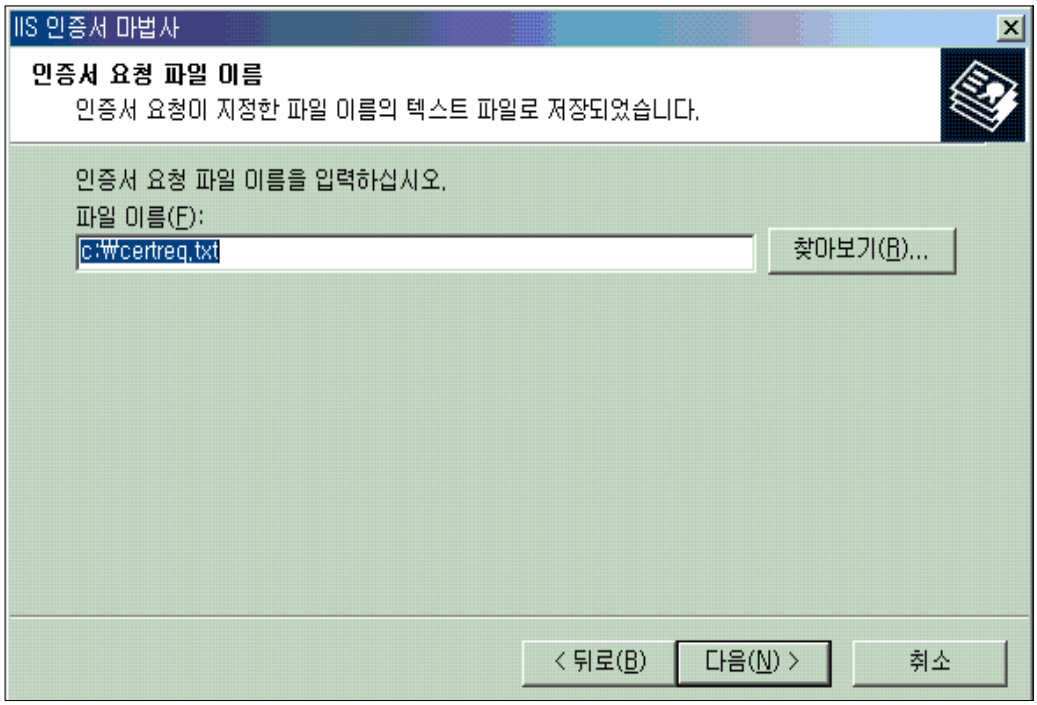
⑦ 인증받을 도메인 이름을 입력하시기 바랍니다.



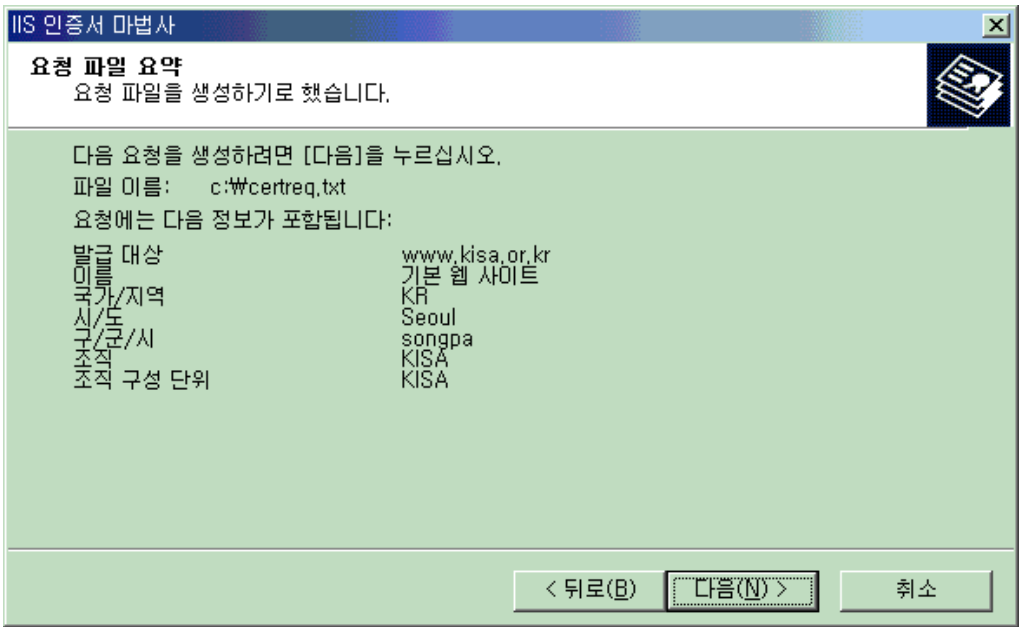
⑧ 지역 정보를 입력합니다.(모든 내용은 영문으로 입력합니다.)



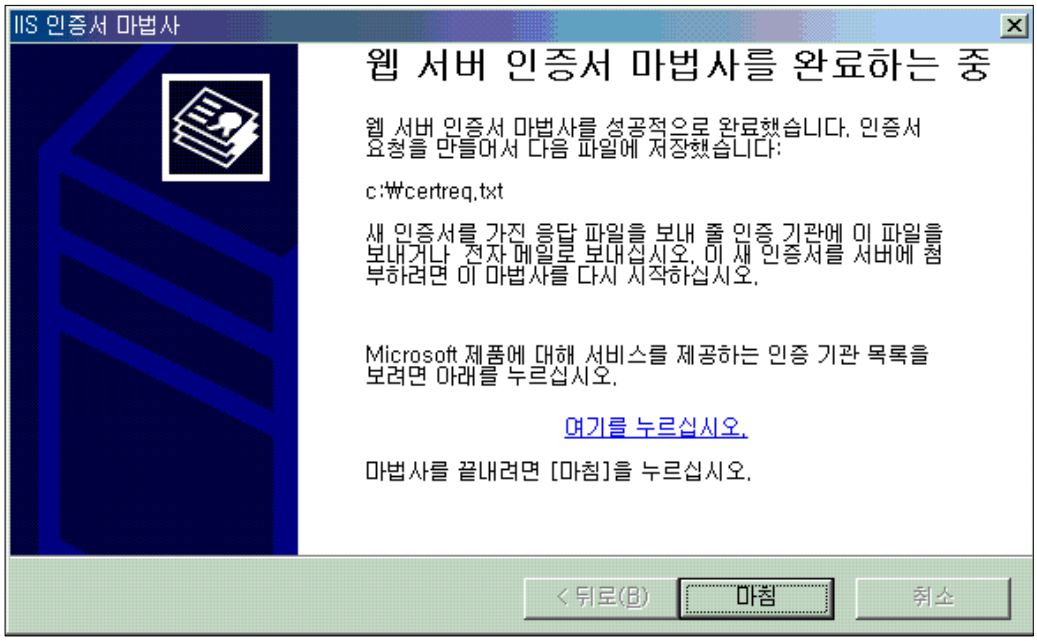
⑨ 인증서 요청파일(CSR)을 저장합니다.



⑩ 신청한 내용을 다시 한 번 확인합니다.



⑪ 인증서 신청을 완료합니다.



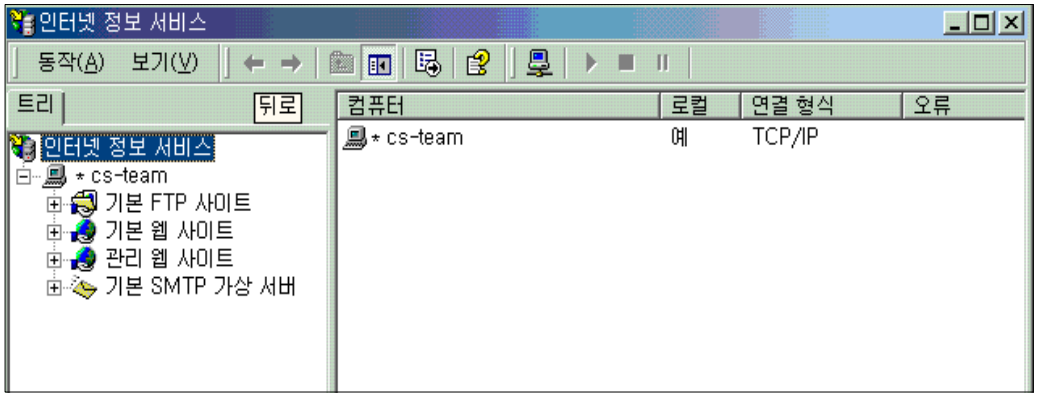
⑫ CSR 내용을 인증기관에게 메일로 송부하시던지 인증서 신청화면에 붙여 넣으신 후 인증서 신청을 진행하시면 됩니다.

자, 이제 인증기관의 발급 절차에 따라서 인증서가 발급됩니다.

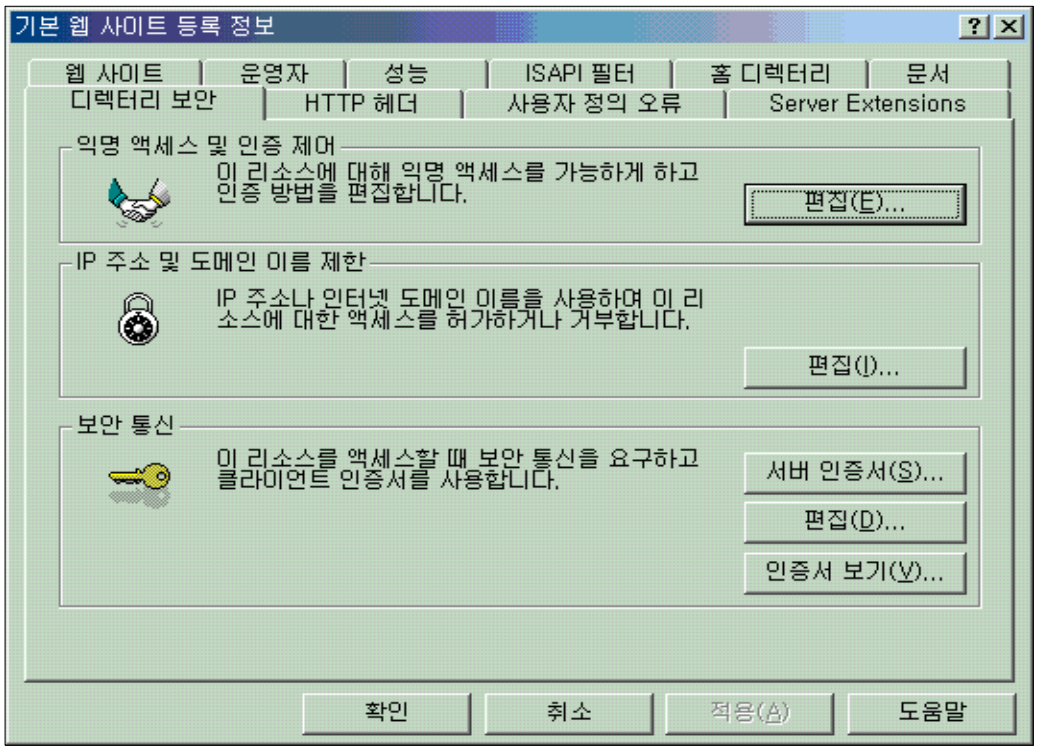
나. SSL 설정

① 웹사이트 속성 선택

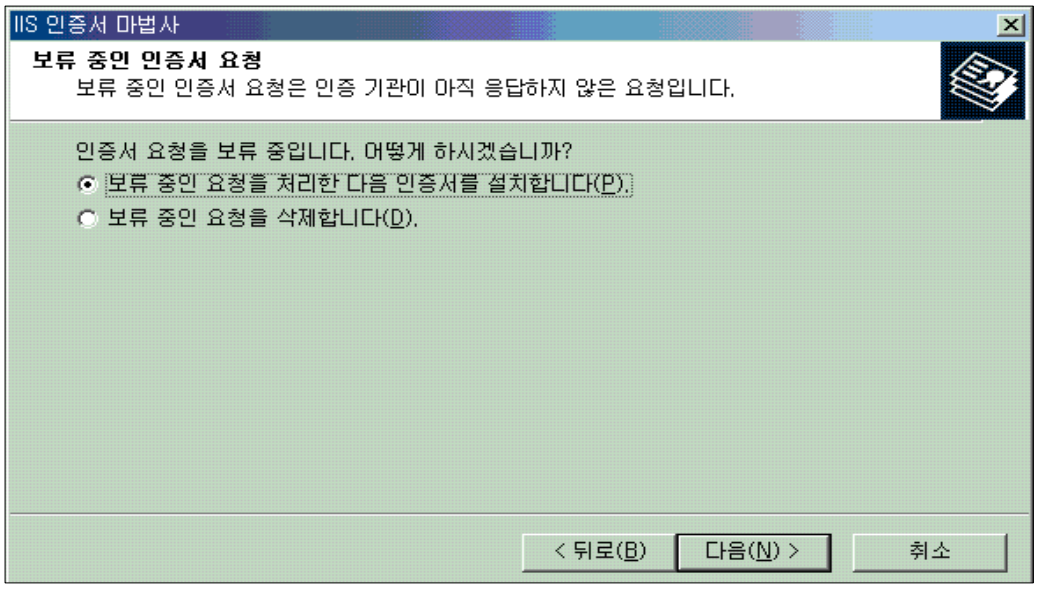
시작→프로그램→관리도구→인터넷 서비스 관리자→웹사이트→속성



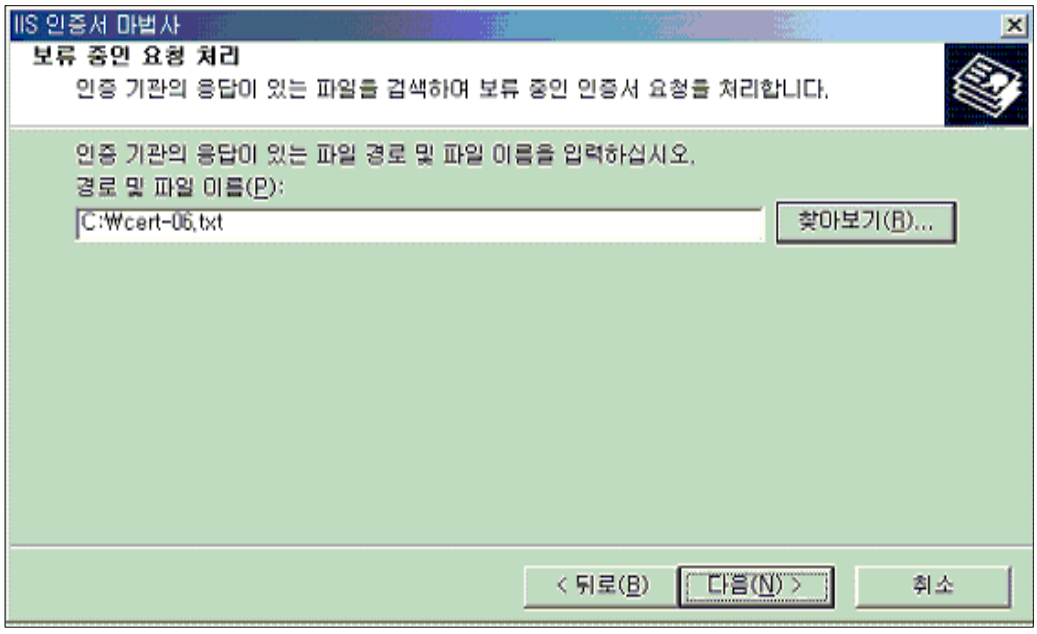
② 등록정보 화면에서 디렉토리 보안을 클릭한 후 서버 인증서를 클릭합니다.



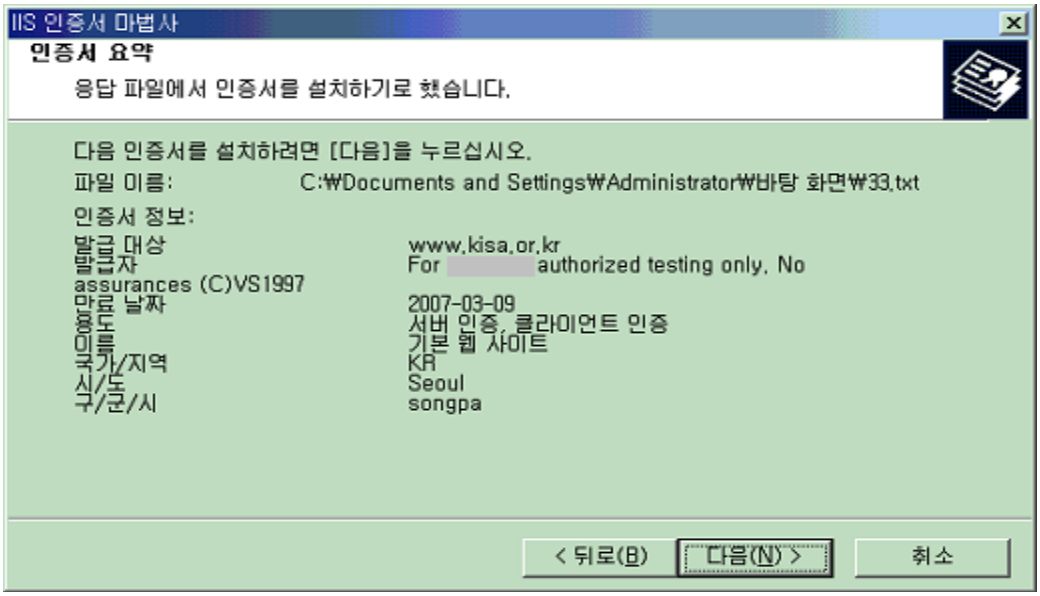
③ 보류중인 요청을 처리합니다.



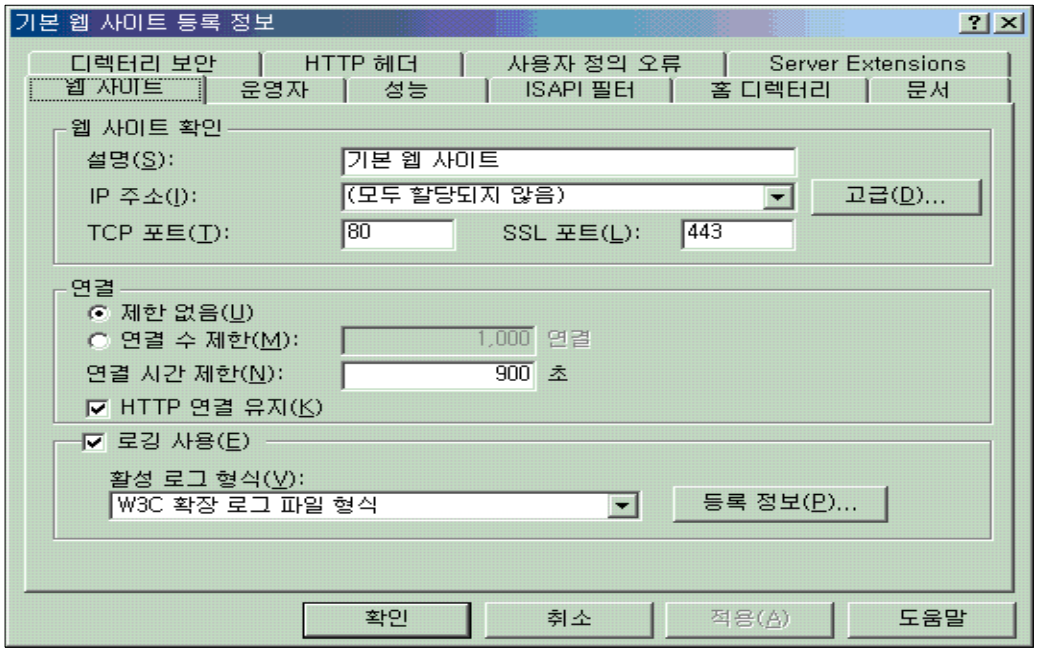
④ 보류 중인 요청 처리-메일을 통하여 받은 인증서(-----begin 부터 end-----까지)를 저장한 파일을 선택합니다. 인증서 파일을 선택한 후 다음 버튼을 누릅니다.



⑤ 인증서 요약 - 현재 설치하시고자 하는 인증서의 내용이 보여집니다. 만약에 신청하신 내용과 일치하지 않으면, 경고 메시지가 뜨며, 인증서가 설치되지 않습니다. 그럴 경우에는 현재의 요청을 삭제하신 후, 새로운 인증서를 신청하셔야 합니다.



⑥ 인증서 설치 후의 설정 - 기본 웹 사이트의 등록정보에서 웹사이트 탭을 선택합니다. 웹 사이트 확인 섹션에서 고급 버튼을 클릭해서 SSL 포트에 443을 설정해줍니다.(기본적으로 443을 사용하지만, 사이트 운영자가 1~65535 범위내에서 임의로 포트번호를 설정할 수 있습니다)



⑦ 인증서 설치 확인 - 인증서가 정확히 설치되었는지 인증서가 설치된 홈페이지를 통해 확인할 수 있습니다.
https://인증서 신청 URL에 접속해서 하단에 노란자물쇠 버튼이 뜨는지 확인합니다. 만일 443이 아닌 다른 포트로 SSL 포트를 적용하였을 경우에는 주소창 뒤에 포트번호를 지정해야 확인할 수 있습니다.(예 : https://www.kisa.or.kr:442)

⑧ 이제 SSL 인증서의 설치가 완료되었습니다. VI장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

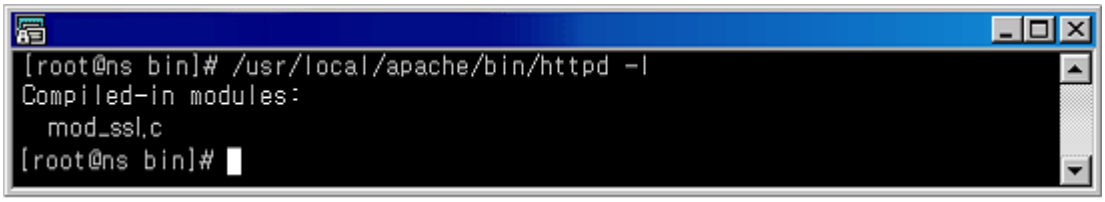
2.2 아파치 서버에서의 설치 과정

가. 아파치 서버에 Openssl과 MOD_SSL의 설치 방법

아파치 서버에서 SSL 통신을 가능하게 하기 위해서는 OpenSSL과 mod_ssl이 필요합니다.

우선, 현재 서비스 중인 Apache 서버에 Mod_SSL이 설치되어 있는지를 httpd -l 옵션을 사용하여 Mod_SSL.c 또는 Mod_SSL.so가 있는지 확인하시기

바랍니다. 만일 설치되어 있다면 '나. 개인키 생성 및 CSR 생성 방법' 과정으로 이동하시기 바랍니다.



<그림 3-3> Mod-SSL 설치 확인 예

OpenSSL은 최신버전으로 설치하는 것을 권장하지만 서버의 환경에 따라서 적합한 버전으로 설치하는 것이 좋습니다. 서버 환경에 맞는 OpenSSL 버전은 www.openssl.org에서 확인할 수 있습니다.

또한 Mod_SSL은 반드시 Apache 버전에 맞는 것을 설치하셔야 하며 www.modssl.org에서 Apache 버전을 확인 한 후 그에 맞는 mod_SSL을 다운받아 설치하시기 바랍니다.

① OpenSSL의 설치(www.openssl.org)

압축풀기

```
$ gzip -cd openssl-0.9.6.tar.gz | tar xvf -
```

$ ./config$ make$ make installconfig

☞ prefix를 주지 않았을 때에는 /usr/local/ssl 디렉토리에 설치가 됩니다.

다른 디렉토리에 설치를 하고자 한다면 다음과 같이 합니다.

$ ./config --prefix=/usr/local --openssldir=/usr/local/openssl

☞ OpenSSL의 실행파일은 /usr/local/ssl/bin에 설치되고 인증서비스를 위한 파일들은/usr/local/openssl 아래의 디렉토리에 생성됩니다.

② mod_ssl의 설치 (www.modssl.org)

압출풀기

```
$ gzip -cd apache_1.3.19.tar.gz | tar xvf
$ gzip -cd mod_ssl-2.8.1-1.3.19.tar.gz | tar xvf
```

파일의 다운로드와 압축풀기가 끝나면 mod-ssl 설정을 합니다.

mod-ssl 설정

```
$ cd mod_ssl-2.8.1-1.3.19
$ ./configure \
--with-apache=../apache_1.3.19 \
--with-ssl=../openssl-0.9.6 \
--prefix=/usr/local/apache
```

③ Apache Server 설치(www.apache.org)

```
$ cd ../apache_1.3.x
$ SSL_BASE=../openssl-0.9.6 \
./configure \
--prefix=/usr/local/apache \
--enable-module=ssl \
$ make
$ make certificate
$ make install
```

나. 개인키 생성 및 CSR 생성 방법

① 랜덤 넘버 생성

```
$ openssl md5 * > rand.dat
```

② 키 쌍 생성

```
$ openssl genrsa -rand rand.dat -des3 -out 1024 > key.pem
```

☞ 개인키 비밀번호를 입력하며 반드시 기억해야 합니다. (암호를 분실할 경우 SSL 사용을 위한 apache를 구동할 수 없습니다)

③ 생성된 키 쌍을 이용하여 CSR생성

```
$ openssl req -newkey key.pem > csr.pem
```

☞ 여기서 key.pem은 단계 ②에서 생성한 키 이름이며 csr.pem은 출력 CSR 파일의 이름입니다.

다음 정보를 입력하라는 메시지가 나타납니다. (모든 내용은 영문으로 작성해야 하며, 아래는 입력 예입니다)

```
Country(국가 코드)  KR
State/province (시/도의 전체 이름)  Seoul
Locality(시,구,군 등의 이름)  Songpa-gu
Organization(회사 이름)  test (이니셜 또는 약자 금지. Full name입력)
Organization Unit(부서명) : team
Common Name (host name + domain name) : cert.kisa.or.kr
```

"추가 속성"을 입력하라는 메시지가 나타나면 그냥 넘어가셔도 무방합니다.

④ CSR 제출

생성된 CSR(예:csr.pem)의 내용은 다음과 같습니다.

```
-----BEGIN CERTIFICATE REQUEST-----
MIIBETCBvAIBADBXMQswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0ZTEh
MB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNAQkB
FgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV1LQG
fcSh8nehEOIxGwmCPlrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwEA
AaAAMA0GCSqGSIb3DQEBBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE REQUEST-----
```

CSR 내용을 인증기관에게 메일로 송부하시던지 인증서 신청화면에 붙여넣으신 후 인증서 신청을 진행하시면 됩니다.
인증기관의 발급 절차에 따라서 인증서가 발급됩니다.

다. 인증서 설치 방법

① 메일로 받은 인증서를 복사하여 파일로 저장합니다.(예: Cert.pem)

```
-----BEGIN CERTIFICATE-----
MIIBETCBvAIBADBXMQswCQYDVQQGEwJBVTETMBEGA1UECBMKU29tZS1TdGF0Z
TEhMB8GA1UEChMYSW50ZXJuZXQgV2lkZ2l0cyBQdHkgTHRkMRAwDgYJKoZIhvcNA
QkBFgFgMFwwDQYJKoZIhvcNAQEBBQADSwAwSAJBAL6nPTy3avNgbubx+ESmD4LV
1LQGfcSh8nehEOIxGwmCPlrhTP87PaA0XvGpvRQUjCGStrlQsd8lcYVVkOaytNUCAwE
AAaAAMA0GCSqGSIb3DQEBBAUAA0EAXcMsa8eXgbG2ZhVyFkRVrI4vT8haN39/QJc9
BrRh2nOTKgfMcT9h+1Xx0wNRQ9/SIGV1y3+3abNiJmJBWnJ8Bg==
-----END CERTIFICATE-----
```

② 아파치 서버에 적절한 위치에 저장합니다.

③ 환경설정 파일(httpd.conf 또는 ssl.conf)을 수정합니다.

```
<VirtualHost _default_:443>

#  General setup for the virtual host
DocumentRoot /Apache/htdocs
ServerName @@ServerName@@:443
ServerAdmin cs@crosscert.com
ErrorLog logs/error_log
TransferLog logs/access_log
SSLCertificateFile /Apache/ssl/cert.pem → 인증서 파일 경로
SSLCertificateKeyFile /Apache/ssl/key.pem → 개인키 파일 경로
```

④ 아파치를 재구동합니다.

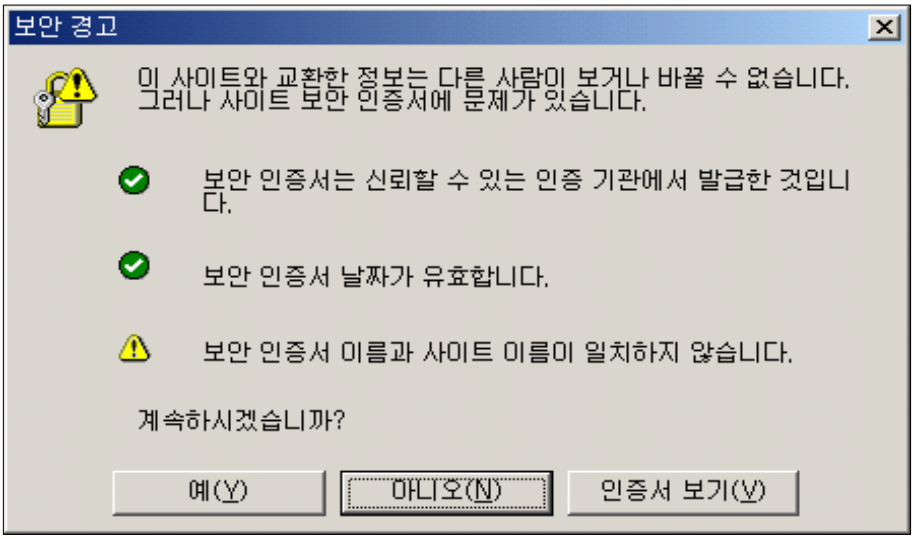
```
./apachectl startssl
```

아파치 서버에서 SSL을 사용하기 위한 시작 명령어인 startssl을 실행하면 개인키의 비밀번호를 묻는데, 이 비밀번호는 이전의 설치과정 중 '나. 개인키 생성 및 CSR 생성 방법' 중 ② 키 쌍 생성시 입력한 개인키 비밀번호를 입력하시면 됩니다.

이제 SSL 인증서의 설치가 완료되었습니다. Ⅵ장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

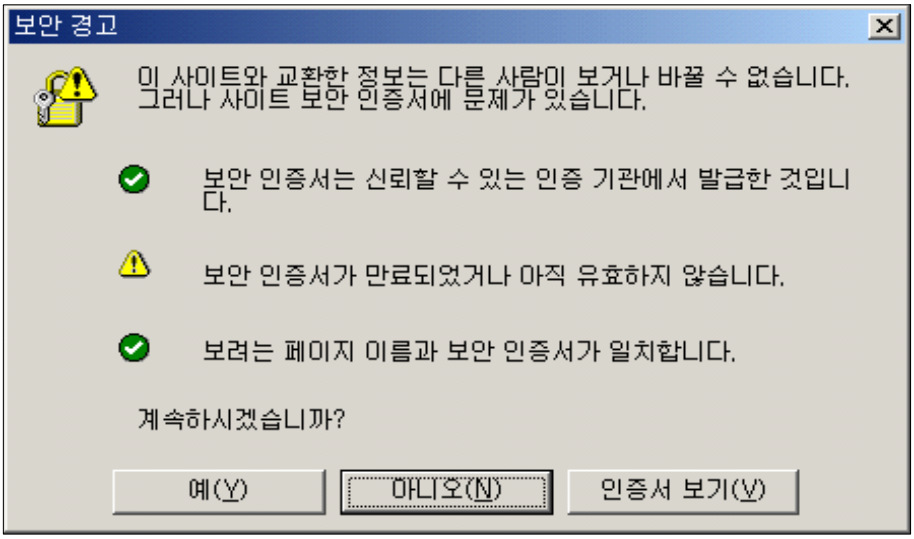
3. 오류 발생시 대처방법

① 인증서를 발급받은 사이트 주소와 실제로 접속한 사이트 주소가 다른 경우



예를 들어 www.kisa.or.kr로 인증서를 발급받아 설치한 후 실제 적용은 login.kisa.or.kr로 걸어두는 경우와 같이 인증서를 발급받은 주소와 실제로 접속한 주소가 다른 경우에 위와 같은 경고창이 나오게 됩니다.

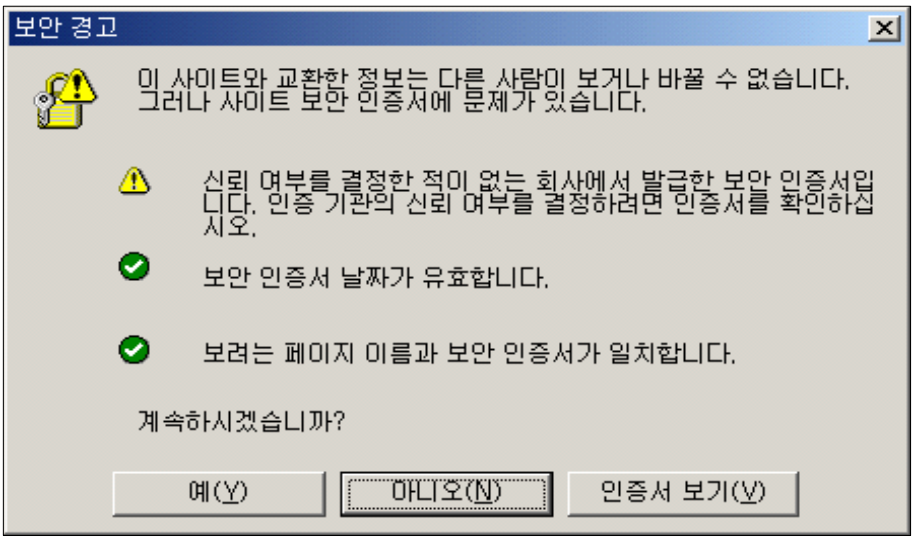
② 인증서가 유효하지 않은 경우



인증서는 저마다 고유한 유효기간을 가지고 있는데, 이 기간이 지난 인증서를 계속 설치해 두는 경우에 나오는 경고창입니다.

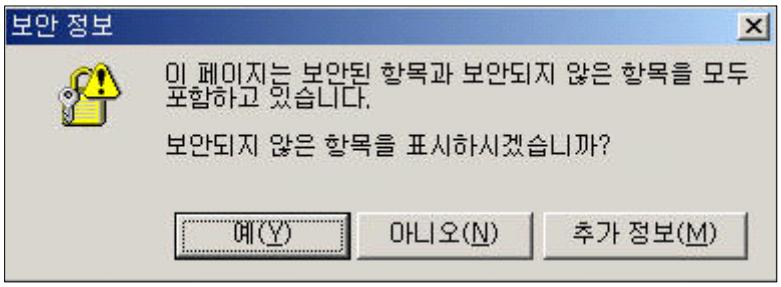
그러나, 보통의 경우 인증서가 설치된 사이트에 접속하는 PC의 날짜가 잘못되어 있어서 생기는 경우가 가장 많습니다.

③ 브라우저가 웹 서버 인증서를 신뢰할 수 없는 경우



이 경우는 웹 서버 인증서를 발급한 인증기관을 웹 브라우저가 인식하지 못하는 경우로써, 브라우저에는 기본적으로 신뢰할 수 있는 인증기관 리스트가 내장되어 있는데 그 리스트에 없는, 즉 신뢰할 수 없는 인증기관에서 발급된 인증서를 설치한 경우에 발생하는 경고창입니다. 실제로는, 웹 서버에서 자체적으로 만든 인증서를 설치한 경우에 가장 많이 생깁니다.

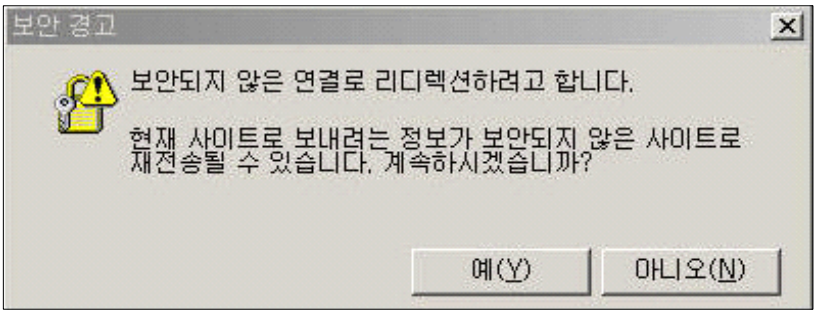
④ 보안된 항목 https와 보안되지 않은 항목 http를 모두 포함하는 경우



말 그대로 보안된 항목 https와 보안되지 않은 항목 http를 모두 포함하고 있어 나타나는 보안경고창입니다. https://를 이용해서 암호화 통신을 하고자 하는 페이지의 소스에 http://를 이용하여

호출하는 이미지 등이 존재할 때 보안경고창이 나타나는 것입니다. 이 경우 '아니오' 버튼을 눌러 표시되지 않는 http 항목의 소스를 절대경로를 써서 https로 호출하시면 됩니다.

⑤ 한 페이지에 http://와 https://의 두 프로토콜이 존재하는 경우



한 페이지 안에 http://와 https://의 두 프로토콜이 존재하기 때문입니다.

예) HTTP://[로그인 폼 입력 페이지]
▽HTTPS://[로그인 서버 처리]
▽HTTP://[로그인 완료 페이지]

예를 들어, http://www.kisa.or.kr에서 https://www.kisa.or.kr/login.jsp로 접속할 때 /login.jsp안에 http://www.kisa.or.kr로 호출하는 직접적인 소스가 있기 때문입니다.

해결방법 : http://리턴URL을 Script를 통해서 Return URL을 설정해 주시면 됩니다.

⑥ 운영중인 웹 서버를 같은 기종으로 변경하려 합니다.

개인키와 인증서를 백업하신 후 재설치하여 사용이 가능합니다. 서버 이전 또는 변경 전 설치 업체에게 반드시 사전 문의 후 작업을 진행하시기 바랍니다.

⑦ 운영중인 웹 서버 종류를 다른 기종으로 변경하려 합니다.

개인키와 인증서를 백업하신 후 재설치하여 사용이 가능합니다. 다만 일부 웹 서버 종류는 인증서 및 개인키의 호환이 안되는 경우가 있으니 서버 이전 또는 변경 전 설치 업체에게 반드시 사전 문의 후 작업을 진행하시기 바랍니다.

⑧ https로 접속하면 페이지를 표시 할 수 없다는 페이지가 보입니다.

이 에러는 아래와 같은 이유로 발생합니다.

ⅰ. https 디렉토리에 파일이 존재하지 않을 경우
ⅱ. 서버의 방화벽이나 end-user의 방화벽에서 443 포트가 차단되었을 경우
ⅲ. https 서버가 다운되었을 경우
ⅳ. SSL Certificate 파일이 정상적이지 않을 경우
ⅴ. 웹 브라우저에서 ssl 3.0으로 셋팅이 되어 있지 않을 경우

인증서가 정상적으로 설치되었는지를 확인하시고 서버에서 https를 위한 포트를 활성화 되었는지 확인하시기 바랍니다. 또한 방화벽과 L4 스위치 등 보안장비가 있다면 https를 위한 해당 포트를 모두 허용하여 주시기 바랍니다. IIS 서버의 경우 'Netstat -na | findstr 포트번호', 아파치 서버의 경우 'Netstat -na | grep 포트번호' 명령어를 이용하여 https를 위한 포트가 활성화되어 있는지 확인할 수 있습니다. 위의 모든 내용을 확인한 후에도 정상적으로 동작하지 않을 경우 해당 업체에게 문의하시기 바랍니다.

⑨ CSR 생성이 정상적으로 되지 않습니다.

CSR을 생성하실 경우 해당 정보는 모두 영문으로 작성하여 주셔야 하며 특수문자는 사용하시면 안됩니다. 또한 입력을 요청하는 모든 내용을 입력하시고 다시 한 번 작업을 하시기 바랍니다. 위의 모든 내용을 확인 한 후에도 정상적으로 동작하지 않을 경우 해당 업체에게 문의하시기 바랍니다.

⑩ 아직 도메인이 없는데 IP를 대상으로 인증서 발급이 가능합니까?

> 인증서는 고유의 식별자를 대상으로 발급됩니다. 즉 개인은 주민등록번호를 기준으로 발급되며 법인은 사업자등록번호로 발급되고, 서버는 도메인을 기준으로 발급되기 때문에 IP 또는 서버이름으로는 발급되지 않습니다. 따라서 도메인을 등록하시거나 서버의 호스트 이름으로 인증서를 신청하셔야 합니다.

⑪ 서브(Secondary) 도메인에 대해서도 보안서버를 적용하고 싶습니다. 어떤 방법이 있을까요?

> 일반적인 보안서버 인증서는 Host*Domain 단위로 발급되고 적용되게 됩니다. 다른 Host*Domain에 대해서도 보안을 적용하고자 한다면 전문업체와 협의하여 적합한 제품을 구축하셔야 합니다.

⑫ SSL 인증서를 발급 받았는데 도메인의 IP를 변경해도 괜찮은가요?

> 인증서는 특정 IP로 제한하여 발급되지 않으며, www.test.co.kr처럼 도메인 이름으로 발급이 됩니다. 따라서 IP를 변경해도 무관합니다.

⑬ 인증서 기간이 만료되어 새로 발급받으려고 하는데 CSR 파일을 전에 사용하던 것으로 가능한가요?

> 보통 웹 서버에서 예전에 사용하던 CSR 파일의 사용이 가능합니다. 그러나 보안상의 이유로 추천하지 않습니다. 인증서를 갱신하실 때마다 새로이 CSR 파일을 생성하시는 것이 좋습니다. 또한 웹 서버 종류마다 반드시 CSR을 생성해야 하는 서버가 있으니 새로 발급받으시기 전에 전문업체에게 문의하시기 바랍니다.

⑭ 보안서버가 구축되었는지 확인을 어떻게 하나요?

일반적으로 보안서버가 구축되어 있으면 https://URL로 확인이 가능합니다. 하지만 https://URL을 통해서도 확인이 되지 않는 경우가 있으니 전문업체에게 확인을 하시면 보다 정확하게 진단을 받으실 수 있습니다.

⑮ 우리가 사용하고 있는 서버에 보안서버 설치가 가능하나요?

기본적으로 모든 웹 서버에는 인증서 방식의 보안서버가 설치 가능하지만 서버 환경에 따라서 별도의 작업이 필요할 수도 있습니다. 보안서버 신청 및 설치 전에 전문업체에게 문의하신 후 작업을 진행하시기 바랍니다.
반드시 확인이 필요한 서버는 Apache, Tomcat 등이 있습니다.

⑯ 인증서를 설치하면 보안서버 구축이 완료되나요?

인증서를 설치하면 보안통신을 위한 기초적인 작업이 완료됩니다. 하지만 실제로 보안통신을 하기 위해서는 인증서 설치 후 웹페이지를 수정하여 https 프로토콜을 호출하는 작업이 필요합니다.
https를 호출하는 작업이 완료되어야만 보안서버가 완전하게 구축, 운영될 수 있습니다. 각 작업은 프로그램 소스를 변경하여 주시는 작업이므로 'Ⅵ장 웹페이지 수정 및 적용 확인하기' 내용을 참고하여 사이트 운영자가 직접 하셔야 합니다.

Ⅳ. 응용프로그램 방식 보안서버 구축하기

1. 소개 및 보안서버 구축 절차

가. 개요

응용프로그램 방식의 경우 SSL 방식과는 달리 제공되는 솔루션마다 설치 방법과 사용방법이 서로 다릅니다. 따라서 본 절에서는 공통적인 설치 과정과 응용프로그램 방식만이 가지는 특징에 대해서 설명하겠습니다.

응용프로그램 방식은 SSL 방식과 같이 클라이언트와 서버간의 데이터를 암호화하는 기능을 제공하고 있습니다. SSL 방식이 웹 브라우저와 웹 서버가 기본적으로 가지고 있는 SSL 기능을 이용하는 것과는 달리 응용프로그램 방식은 웹 브라우저와 웹 서버에 별도의 모듈을 추가하여 데이터를 암호화하는 기능을 제공합니다.

대부분의 웹 브라우저는 기능을 향상시키기 위한 나름대로의 확장기능을 제공합니다. 인터넷 사용시 볼 수 있는 플래쉬나 미디어플레이어 등이 이러한 확장기능을 이용하는 대표적인 예라고 볼 수 있습니다.

응용프로그램 방식은 웹 브라우저와 웹 서버가 기본적으로 제공해 주지 않는 추가적인 기능으로 동작하기 때문에 이러한 확장기능을 통하여 기능을 제공합니다.

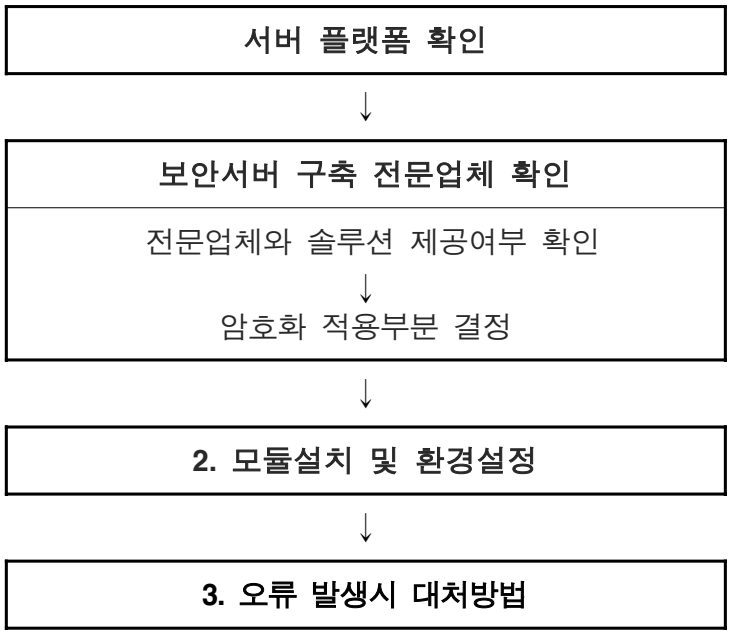
웹 브라우저 확장기능을 이용해 만든 것을 클라이언트 모듈이라고하고, 웹 서버의 확장기능을 이용해 만든 것을 서버 모듈이라고 합니다. 클라이언트 모듈은 일반적으로 Active-X 형태로 제공되며, 서버 모듈은 Java의 경우 Java Servlet Filter 혹은 Jar 파일로 제공되며, IIS의 경우 Filter 혹은 DLL 형태로, PHP의 경우 확장 모듈 형태로 제공됩니다.

나. 보안서버 구축 절차

응용프로그램 방식은 웹 서버에 서버 모듈을 설치해야 하고, 웹 브라우저에 클라이언트 모듈을 다운로드시켜 설치해야 하는 방식이므로 SSL 방식과는 다른 절차가 필요합니다.

SSL 방식의 경우 웹 서버에 인증서를 설치한 후 설정 부분을 수정하고, 실제 사용할 부분의 URL 부분을 수정해 주어야 하는 것과는 달리 응용프로그램 방식은 웹 서버 혹은 웹 어플리케이션 서버에 모듈을 설치하는 과정이 필요하므로 구축 전문업체와의 협력을 통해 설치가 진행됩니다.

일반적인 응용프로그램 방식의 구축 절차는 다음과 같습니다.

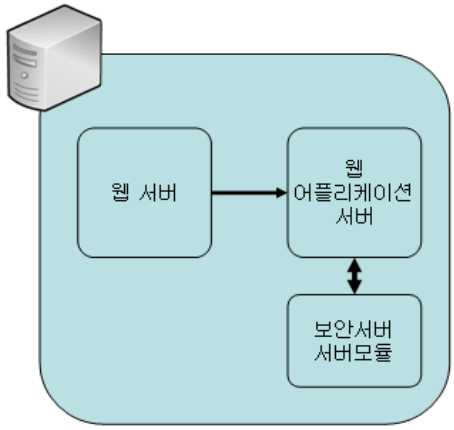


<그림 4-1> 응용프로그램 방식 보안서버 구축 절차

① 서버 플랫폼 확인

보안서버를 응용프로그램 방식으로 구축하기로 결정하셨다면 그 다음으로 하셔야 할 일이 서버 플랫폼을 확인하시는 단계입니다. 응용프로그램 방식은 보안서버 서버모듈을 설치해야 하기 때문에

서버 플랫폼으로 어떤 것을 사용하느냐에 따라 각각에 맞는 서버모듈을 제공하게 됩니다.



<그림 4-2> 서버 플랫폼의 구성

서버 플랫폼은 통상 「웹 서버」와 「웹 어플리케이션 서버」로 나누어질 수 있습니다.

보안서버의 서버모듈은 일반적으로 웹 어플리케이션 서버와 연동됩니다. 따라서 어떤 웹 어플리케이션 서버를 사용하느냐에 따라 제공되는 보안서버 서버모듈도 달라지게 됩니다.

웹 서버는 일반적으로 아파치 웹 서버를 많이 사용하고 웹 어플리케이션 서버로는 자바계열의 톰캣, 제이보스, 웹로직 등이 있고, MS 계열의 ASP와 .Net이 있으며, 그 밖에 PHP 등이 많이 사용되고 있습니다.

② 보안서버 구축 전문업체 확인

현재 운영되고 있는 서버플랫폼을 확인하셨다면 보안서버 구축 전문업체 등을 통해 해당 서버 플랫폼에 맞는 응용프로그램 방식의 솔루션을 제공해 줄 수 있는지 확인하셔야 합니다.

③ 암호화 적용부분 결정

보안서버 구축 전문업체가 결정되었으면 보안서버 구축 전문업체와 함께 암호화를 적용할 부분을 결정하셔야 합니다. 응용프로그램 방식에서의 암호화 부분은 텍스트로 전송되는 부분만을 암호화하여, 이미지와 같이 암호화가 불필요한 부분은 제외시킬 수 있습니다.

④ 모듈설치 및 환경설정

암호화 적용 부분까지 결정되었다면 보안서버 구축 전문업체와 함께 서버 모듈을 설치하셔야 합니다. 이와 함께 필요한 환경설정도 진행되게 됩니다. 환경설정 부분은 제공되는 보안서버 전문업체의 솔루션마다 조금씩 다를 수 있습니다.

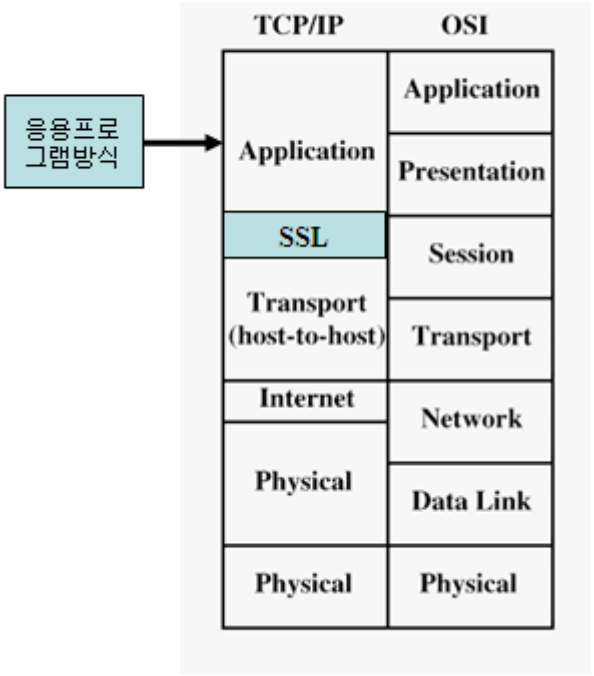
⑤ 구축완료

이제 응용프로그램 방식의 보안서버 구축이 완료되었습니다.

다. 프로토콜 설명

응용프로그램 방식에서 암호화된 데이터를 전달하는 프로토콜은 SSL 방식과는 달리 Application 계층에서 이루어집니다.

SSL 방식의 경우 Application과 Transport 계층 사이에 SSL이라는 별도의 계층으로서 데이터를 전달합니다. 이 방법의 경우 Application에서 암호화할 때 계층을 설정해 주는 일 이외에는 별다른 작업이 필요없는 장점을 가지고 있습니다. 하지만, 계층으로서 데이터를 전송하기 때문에 암호화할 부분을 선택적으로 전송할 수 없어 불필요한 이미지나 동영상과 같은 데이터도 암호화하여 전송할 수밖에 없습니다.



<그림 4-3> 응용프로그램 방식 프로토콜

응용프로그램 방식의 경우는 별도의 계층이 아닌 Application 계층에서 동작합니다. 이러한 방식의 장점은 필요한 부분만 암호화하여 전달할 수 있다는 점입니다. Application에서 필요한 부분만을 적용하여 암호화하여 불필요한 리소스 낭비를 줄일 수 있습니다.

하지만 Application에서 동작하기 위해서는 Application과의 연동 작업이 필요합니다. 즉, 웹 서버쪽 프로그램의 일부의 수정작업이 필요합니다. 응용프로그램 방식을 제공하는 업체에 따라 수정작업의 범위나 양은 많은 차이가 있습니다. 일부 응용프로그램 방식의 경우 SSL 방식과 유사한 형태의 암호화 방식을 취하면서도 선택적 암호화가 가능한 기능을 제공하는 제품도 있습니다.

2. 설치 과정

가. 클라이언트 모듈 설치

앞서 설명된 것처럼 응용프로그램 방식에서는 클라이언트 모듈의 설치가 필요합니다. 클라이언트 모듈이 설치되는 것은 사용자 PC의 웹 브라우저이지만 이를 위해서는 사용자가 웹 서버에 접속했을 때 클라이언트 모듈을 다운로드하여 설치하게끔 하는 사전작업을 수행해야 합니다.

이것은 웹 포털이나 웹 서비스에 접속했을 때 해당 사이트에 플래쉬가 있을 경우 자동으로 플래쉬 구동 프로그램이 설치되는 것과 동일한 원리입니다.

이를 위해서는 웹 서버의 특정 부분에 클라이언트 모듈을 다운로드 및 설치하는 object 태그를 넣어 두어야 합니다.

하지만 이러한 태그를 설치자가 직접 작성하는 것은 아닙니다. 대부분의 응용프로그램 방식을 제공하는 업체들은 별도의 Javascript 파일을 만들어 이 안에서 자동으로 다운로드되고 설치되도록 만들어 두었습니다. 설치시는 이 Javascript 파일을 포함시키면 됩니다.

```
<script language="javascript" src="/abc_plugin.js"> </script>
```

이 Javascript 파일을 포함 시킬 위치는 해당 솔루션을 공급하는 업체와 협의하여 진행하시면 됩니다.

나. 서버 모듈 설치

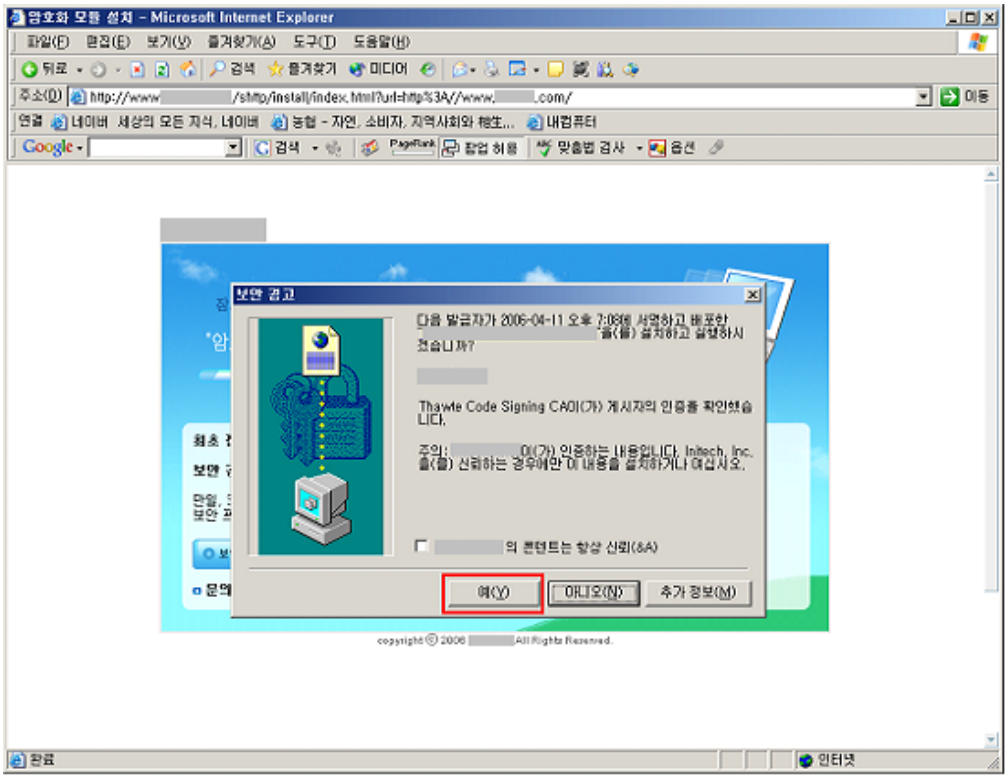
서버모듈의 설치는 응용프로그램 방식을 공급하는 업체별로 또한 사용하시는 웹 어플리케이션 서버의 종류에 따라 많은 차이가 있습니다. 따라서 대부분의 응용프로그램 방식을 제공하는 업체들이 서버모듈 설치 과정을 직접 지원하고 있습니다.

응용프로그램 방식을 사용하고자 한다면 전문업체와 협의하여 해당 웹 어플리케이션 서버에 적절한 서버모듈을 설치하시면 됩니다.

다. 사이트 접속

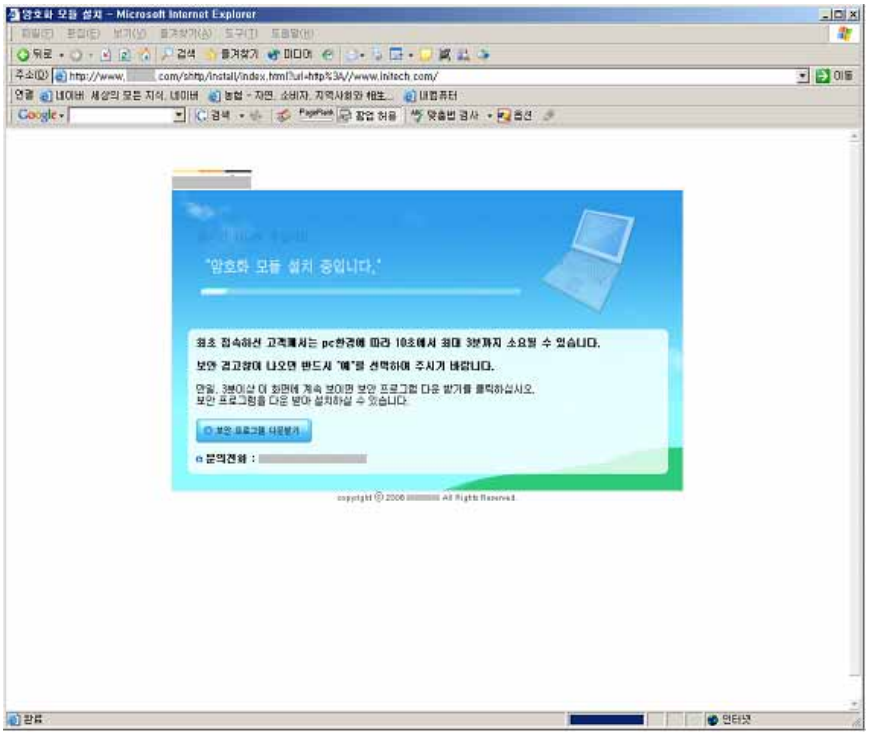
웹 브라우저를 열어 응용프로그램 방식의 보안서버가 구동중인 웹 서버에 접속하게 되면 아래 화면과 같이 클라이언트 모듈 설치에 대한 안내 화면이 나타나게 됩니다.

이 화면은 클라이언트 모듈 설치시 포함시킨 Javascipt 파일에서 자동적으로 동작하도록 구성되어 있기 때문에 필요시 내용을 수정하실 수 있습니다.



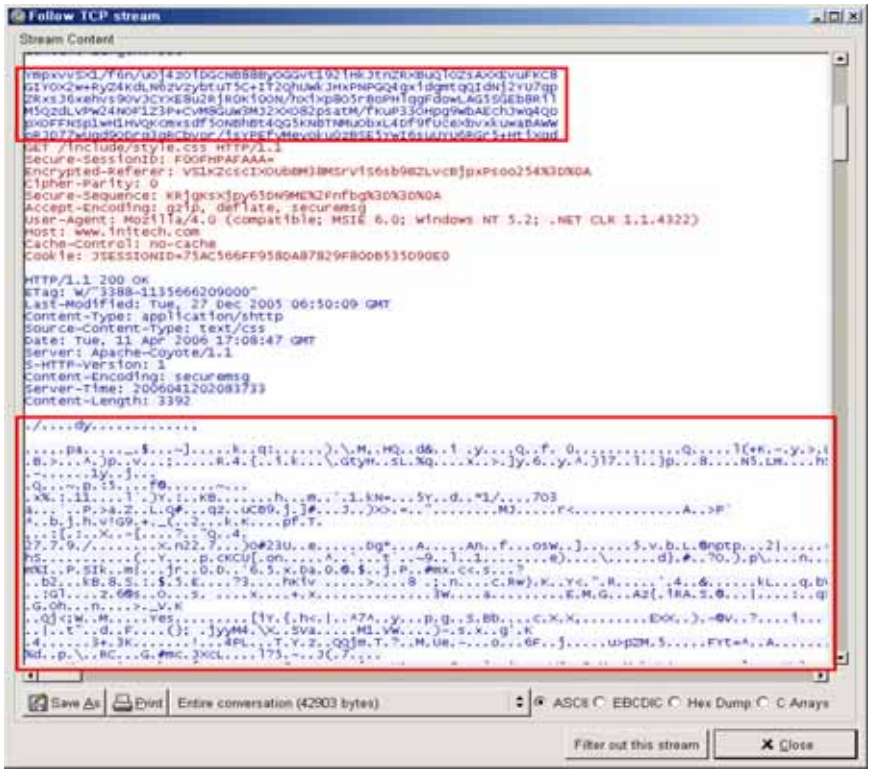
<그림 4-4> 암호화 모듈 설치를 위한 보안경고창

사용자가 설치에 동의할 경우 다음 그림과 같이 자동적으로 클라이언트 모듈이 설치 및 로딩된 후에 사이트가 열립니다.



<그림 4-5> 암호화 모듈 설치

다음은 패킷 캡쳐를 통해 암호화 통신이 되는 것을 확인한 것입니다.



<그림 4-6> 암호화 통신 확인

3. 오류 발생시 대처방법

① 클라이언트 모듈은 Windows 95 등 오래된 OS에서도 설치되나요?

대부분의 응용프로그램 방식 제공업체들은 상당기간에 걸친 기술 개발을 통해 Windows 95 버전 이상 대부분의 OS에서 설치 문제가 발생하지 않는 클라이언트 모듈을 제공하고 있습니다. 하지만 일부 고객들의 PC 이상으로 설치되지 않는 문제가 발생하는 경우 응용프로그램 방식 제공업체들의 고객지원 센터를 통해 기술 지원을 받으실 수 있습니다.

② MS Windows 이외의 OS 및 Firefox 등의 웹 브라우저에서도 클라이언트 모듈을 사용할 수 있나요?

일반적으로 클라이언트 모듈은 MS Windows의 Internet Explorer 브라우저용을 기본적으로 제공합니다. 일부 응용프로그램 방식 제공업체의 경우 Firefox 및 리눅스, 맥용 클라이언트 모듈도 제공하고 있습니다.

③ 응용프로그램 방식이 SSL 방식과 비교되는 장단점은 무엇인가요?

응용프로그램 방식의 경우 앞서 설명된 바와 같이 암호화할 부분을 선택할 수 있어서 암호화에 대한 서버의 부담을 줄일 수 있습니다. 또한 SSL방식에서는 기본적으로 제공하는 암호화 알고리즘 이외에 추가적인 암호화 알고리즘(국산 암호화 알고리즘인 SEED, ARIA 등)을 이용할 수 있습니다.
또한 공인인증서를 이용한 사용자 인증이나 전자서명과 같은 업무를 추가하신다면 응용프로그램 방식에 추가적인 지원이 가능합니다. 하지만 서버 모듈을 제공하여야 하고 상당부분의 기술지원이 필요하기 때문에 연간 서비스 형태로 제공되는 SSL에 비해서 구축비용이 많이 소요될 수 있습니다.

④ SSL 방식은 HTTP 80 포트와는 별도로 443 등의 보안 포트를 사용합니다. 응용프로그램 방식은 어떻습니까?

> 앞서 설명드린대로 응용프로그램 방식은 Application 계층에서 사용되기 때문에 HTTP 80 포트를 그대로 사용합니다.
> 하지만, 일부 응용프로그램 방식의 경우 제품의 특성상 별도의 포트를 사용하는 제품도 있으니 전문업체에 확인하시기 바랍니다.

⑤ 파일을 암호화하여 업로드하고 싶은데 응용프로그램 방식에서도 가능한지요?

> 응용프로그램 방식 제공업체에서는 별도의 추가 모듈로서 파일을 암호화하여 업로드하는 기능을 제공하고 있습니다. 솔루션 제공여부는 제공업체에 문의하시기 바랍니다.

Ⅴ. 웹호스팅업체의 보안서버 구축하기

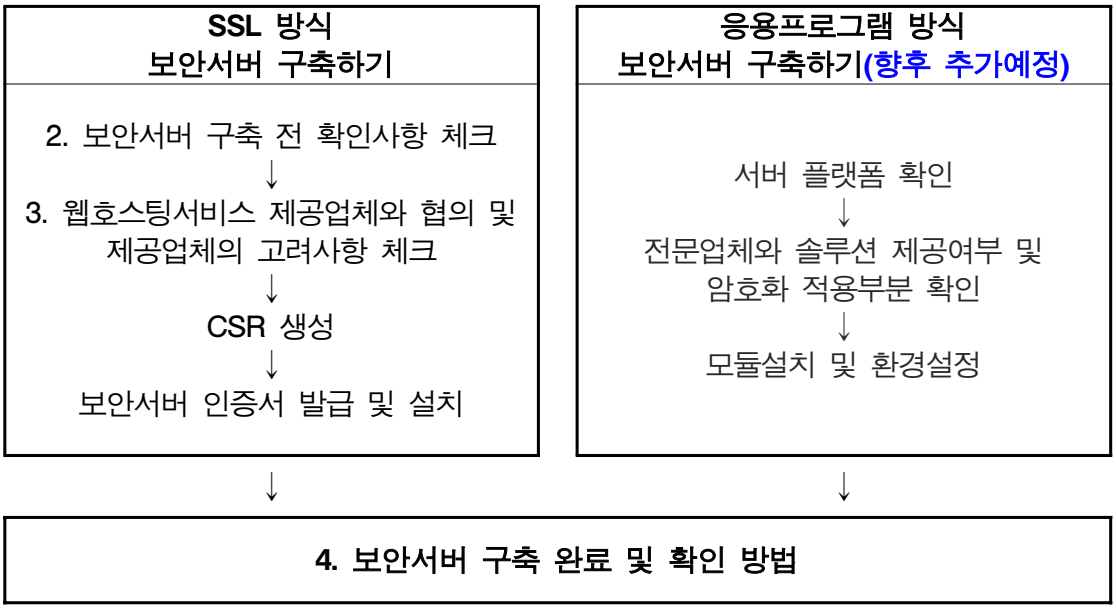
1. 보안서버 구축 절차

단독 서버가 아닌 웹호스팅 서비스를 받고 있는 사이트의 경우, 직접 웹 서버를 수정할 권한을 가지고 있지 않으며, 보안서버 구축을 위한 기술력을 갖추지 않은 경우가 대부분이기 때문에 웹호스팅 서비스 제공업체와의 협의를 통하여 보안서버를 구축하게 될 것입니다.

따라서 원활한 보안서버 구축을 위하여 본격적인 구축 이전에 확인할 사항과 보안서버 구축 이후에 암호화 적용 여부를 확인할 수 있는 방법을 중심으로 설명하고자 합니다.

그리고 웹호스팅 서비스 제공업체 또한 고객사에서 보안서버 구축을 요청할 경우 서버환경과 서비스 제공 가능성 확인 등 고려해야 할 사항을 정리하였습니다.

보안서버 구축은 각 업체마다 구체적인 환경에 따라 달라지지만, 일반적으로 다음과 같은 절차를 따르게 됩니다. 이번 가이드에서는 웹호스팅업체들이 주로 사용하고 있는 SSL 방식의 보안서버 구축 방법을 알아보고, 향후 응용프로그램 방식의 보안서버 구축 방법을 가이드에 추가하도록 하겠습니다.



<그림 5-1> 웹호스팅업체의 보안서버 구축 절차

2. 보안서버 구축 전 확인사항 체크

가. 발급 도메인에 대한 정보 확인

보안서버 인증서 발급시 사전 정보를 확인하는 절차로 인증서를 신청한 업체마다 조금씩 상이한 방식으로 정보를 확인할 수 있습니다.

① 보안서버 인증서 신청정보 확인

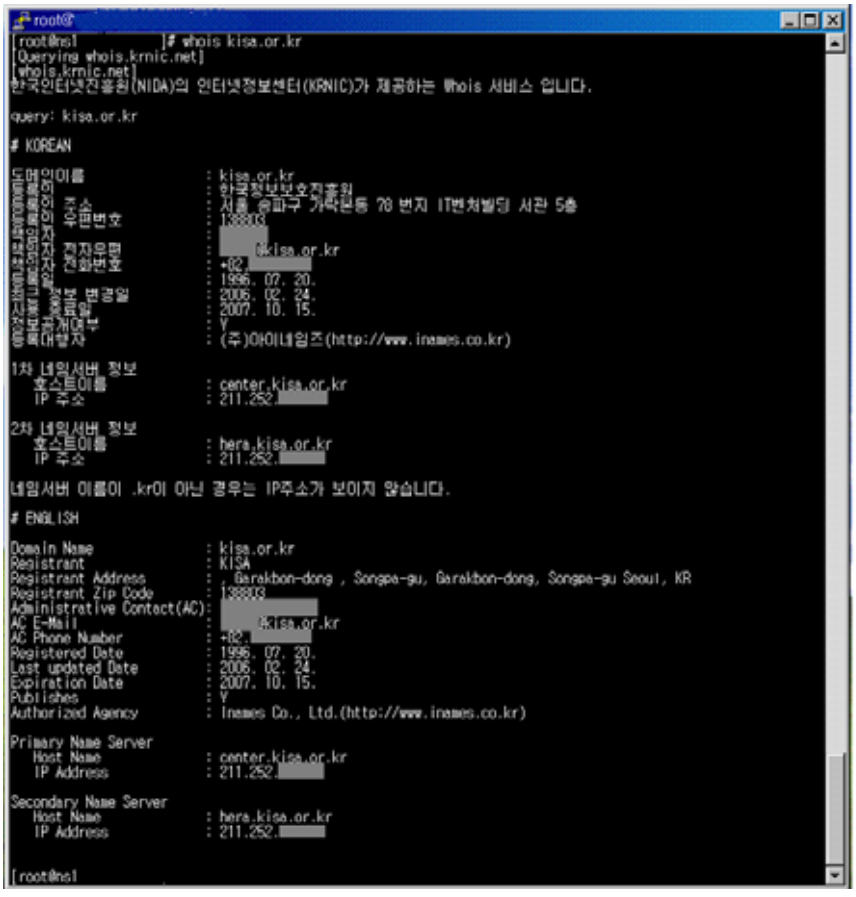
보안서버 인증서를 신청하기 전에 사업자 정보의 주소나 연락처 등을 현재 운영중인 내용으로 수정하여 주시기 바랍니다.

정보 확인은 사용중인 URL 확인과 실제 서비스를 하고 있는 업체를 구별하여 유령회사나 그와 유사한 업체를 사칭하여 인터넷을 이용한 범죄행위를 방지하기 위한 방법입니다.

② 신청사 도메인 정보 확인

도메인 소유 확인을 통해 확인한 회사명과 소유자의 사업자 등록증상의 회사명과 동일한지 확인합니다. 동일하지 않다면 소유한 도메인 정보를 변경하여야 합니다. 일부 인증기관의 경우 신청사 도메인의 등록정보와 신청사의 웹 서버 정보가 다를 경우 인증서 발급이 되지 않을 수 있으니 사전에 확인하여야 합니다.

인증서를 신청하기 위해서는 기본적으로 도메인을 소유하고 있어야 합니다. 또한 인증서는 웹서비스의 실재성, 즉 도메인 소유자가 직접 서비스를 운영하는지, 운영자가 실존하는 단체나 회사인지를 확인하여 발급됩니다. WHOIS을 통해 보안서버 인증서 신청 도메인명에 대한 소유권을 미리 확인하시기 바랍니다.



<그림 5-2> WHOIS를 통한 도메인 정보 확인

③ 신청사 권한 확인

인증기관은 보안서버 인증서를 발행하기 전에 인증서 신청업체가 등록 요청에 지정한 이름으로 사업을 할 수 있는 법적 권한이 있는지 여부를 확인합니다. 확인 절차 및 방법은 인증기관이 요구하는 절차에 따라 진행하시면 됩니다.(예 : 사업자 등록증, 대표자의 신분증 사본, 전화번호 영수증 등의 발송 요구)

경우에 따라서 인증기관의 확인절차 중 영문 사업자 등록증을 제출해야 경우가 있습니다. 영문 사업자등록증의 경우 관할세무서를 방문하시면 발급이 가능합니다. 관할 세무서에 대한 정보는 국세청 홈페이지(http://www.nts.go.kr)에서 확인할 수 있습니다.

나. CSR 생성 및 보안서버 적용

CSR 생성 및 보안서버 적용에 대한 구체적인 절차는 'Ⅲ장 SSL 방식 보안서버 구축하기' 내용을 참고하여 작업하시기 바랍니다.

3. 웹호스팅서비스 제공업체의 고려사항

웹호스팅 서비스를 이용하고 있는 고객사로부터 보안서버 구축에 대한 의뢰가 들어오면 웹호스팅 서비스 제공업체는 다음과 같은 사항들을 고려하여 보안서버를 구축해야 합니다.

가. 서비스 제공 서버에서 개별 인스턴스로 서비스가 가능한지 여부

① 서버가 IIS 계열인 경우

인증서는 1개의 인스턴스에 1개의 인증서만이 설치가 가능합니다. 따라서 한 개의 인스턴스에서 호스트헤더 분리를 사용하여 여러 도메인을 서비스할 경우에는 인증서 서비스가 필요한 사이트를

개별 인스턴스 사이트로 분리해야 합니다.

② 서버가 Apache 계열인 경우

Apache 서버에서는 VirtualHost를 한 개의 개별 인스턴스로 인식합니다. Apache 서버는 일반적으로 VirtualHost를 사용하여 여러 도메인을 서비스하고 있기 때문에 큰 무리없이 진행하실 수 있습니다.

나. SSL 보안 포트 서비스 가능 여부

일반적으로 한 개의 IP에서는 SSL을 위한 보안포트가 중복되어 사용할 수 없습니다. 따라서 한 대의 서버에서 여러 도메인에 대한 SSL 보안을 서비스하시려면 보안 서비스를 제공하는 도메인 수만큼의 보안포트의 확보가 필요합니다.

다만, 멀티도메인 SSL 인증서를 사용하면 한 개의 IP에서 보안포트를 중복하여 사용할 수 있습니다. 보다 자세한 내용은 '부록 A. 멀티도메인 SSL 인증서 소개' 내용을 참조하시기 바랍니다.

보안포트가 확보되면 각 보안포트에 대해서는 방화벽 등의 보안장비에서 각 보안포트에 대해서 접근 허용 정책을 추가해야 합니다. 이는 각 서비스 업체의 보안정책과 관련된 사항이므로 사전에 충분히 검토하시기 바랍니다.

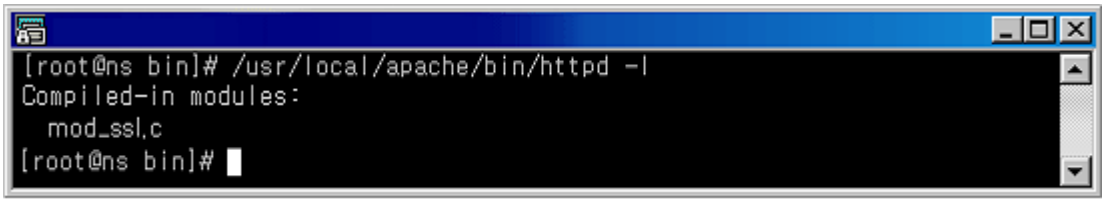
다. SSL 서비스 가능 여부

① 서버가 IIS 계열인 경우

IIS 계열은 기본적으로 모두 SSL 서비스를 제공하도록 구성되어 있습니다.

② 서버가 Apache 계열인 경우

Apache 서버에서 SSL 서비스를 제공하기 위해서는 반드시 Mod_SSL이 설치되어 있어야 서비스 제공이 가능합니다.
현재 서비스 중인 Apache 서버에 Mod_SSL이 설치되어 있는지를 httpd -l 옵션을 사용하여 Mod_SSL.c 또는 Mod_SSL.so 가 있는지 확인하고, 만일 Mod_SSL이 설치되어 있지 않다면 'Ⅲ장 SSL 방식 보안서버 구축하기 중 아파치 서버에서의 설치 과정'을 참고하여 설치하시기 바랍니다.



<그림 5-3> Mod-SSL 설치 확인 예

라. 인증서 신청하기

SSL 방식의 보안서버 인증서는 인증서의 소유가 호스팅 업체가 아닌 사이트 운영자가 소유하게 됩니다. 따라서 인증서를 신청을 하실 경우에는 서버를 소유하거나 운영 대행하는 업체의 정보가 아닌 실제 도메인을 소유하고 서비스를 제공하는 주체의 정보가 필요합니다.

① CSR 생성시 입력 정보
모든 정보는 영문으로 입력하셔야 하며 특수문자를 사용하시면 안됩니다. 다음의 예시를 참고하시기 바랍니다.

CN=운영중인 URL
OU(조직구성단위)=영문 부서명
O(조직명)=도메인을 소유하고 서비스를 제공하는 회사의 영문 전체이름
L(구/군)=도메인을 소유하고 서비스를 제공하는 회사의 위치
S(시/도)=도메인을 소유하고 서비스를 제공하는 회사의 위치
C(국가코드)=KR(대한민국인 경우)

② 준비 서류

공통적으로 도메인을 소유하고 서비스를 제공하는 회사의 사업자 등록증을 준비해야 하며, 인증기관에 따라서 도메인 소유에 대한 확인 절차 및 전화 확인 절차가 있는 경우가 있습니다. 이런 경우에는 추가적인 서류가 필요할 수 있습니다.(예: 도메인 소유 확인증, 전화번호 영수증, 영문 사업자 등록증 등)

4. 보안서버 구축상태 확인

인증서 설치가 완료된 후 정상적으로 인증서가 발급되어 동작되는지 여부는 다음과 같은 방법으로 식별할 수 있습니다.

① 웹 브라우저에 도메인의 URL의 주소를 http가 아닌 'https'를 통해 연결을 시도합니다.

② SSL 인증서버의 경우 기본적으로 80번 포트가 아닌 보안포트를 사용하게 됩니다.

https를 이용하여 접속하시면 보안 포트(기본 443번)로 연결이 되어 별다른 작업없이 SSL 인증서버를 통한 통신을 하게 됩니다.

③ 화면 아래에 잠겨진 자물쇠 아이콘이 있습니다.

보다 자세한 내용은 'Ⅵ장 웹페이지 수정 및 적용 확인하기'에서 다루겠습니다.

자, 이제 SSL 인증서의 설치가 완료되었습니다. Ⅵ장으로 이동하셔서 실제 웹페이지를 어떻게 수정해야 하는지 알아보겠습니다.

Ⅵ. 웹 페이지 수정 및 적용 확인하기

1. 웹 페이지 수정 방법 및 사례

암호화 통신을 하기 위해서 보안 프로토콜을 호출하는 방법은 OS나 Program 언어를 가리지 않고 모두 동일합니다. 그 이유는 암호화 통신을 하기 위해 적용하는 부분이 특정 OS나 특정 Program 언어에 의존하지 않는, 모두가 공통으로 사용하는 HTML 언어이기 때문입니다.

본 절에서는 암호화 적용 범위에 따라 웹페이지 전체 혹은 일부를 암호화하는 방법과 이용자가 선별적으로 암호화를 선택하는 방법을 소개하겠습니다.

가. 전체 페이지 암호화하기

① https 프로토콜 호출하기

https 프로토콜을 호출하여 웹페이지 전체에 적용하는 방법은 그림만으로도 곧바로 이해를 할 수 있을 정도로 아주 쉽습니다. 간단히 호출하는 프로토콜을 http://에서 https://로 수정해 주기만 하면 됩니다.

<그림 6-1>과 <그림 6-2>는 암호화 통신을 하기 위해 https 프로토콜을 호출하기 전과 호출한 후의 HTML 소스코드 예입니다.

```
if ($time3 == $time4) {
echo "
<p><a href='http://[illegible].co.kr/zboard/view.php?id=not
desc=asc&no=$no' target='_top'><font size=1 color='silv
:new::-></a></p> ";
} else {
echo "<p><a href='http://[illegible].co.kr/zboard/view.php?
adnum&desc=asc&no=$no' target='_top'><font size=1 color
}
```

<그림 6-1> 평문 통신을 위한 HTML 소스코드

```
if ($time3 == $time4) {
echo "
<p><a href='https://[illegible].co.kr/zboard/view.php?id=noti
&desc=asc&no=$no' target='_top'><font size=1 color='silve
::new::-></a></p> ";
} else {
echo "<p><a href='https://[illegible].co.kr/zboard/view.php?i
eadnum&desc=asc&no=$no' target='_top'><font size=1 color=
}
```

<그림 6-2> https 프로토콜을 호출하기 위한 HTML 소스코드

② 리다이렉션(Redirection) 설정

앞서 설명을 하였듯이, 암호화 통신을 위해서는 https 프로토콜을 직접 호출을 해줘야 합니다. 하지만, 웹페이지에 접속하는 사용자들은 일일이 https 프로토콜을 붙여서 입력을 하지 않습니다. 대부분의 경우가 www.test.co.kr 또는 test.co.kr 도메인을 웹 브라우저의 주소창에 입력하고 접속하는 경우가 대부분일 것입니다. 이 때 웹 브라우저에 그냥 도메인주소만 입력하면, 웹 브라우저는 해당 도메인 앞에 http://가 붙은 것으로 판단하고 평문 통신을 하도록 합니다.

평문 통신을 하는 경우라면 문제가 없지만, 암호화 통신을 해야 할 경우에는 https://를 직접 붙여서 입력해야 하므로 여간 불편해 하지 않습니다. 리다이렉션은 현재 접속한 도메인이나 혹은 웹페이지를 강제로 다른 주소나 다른 페이지로 변경해 줌으로써 사용자들의 불편함을 감소시켜주고 자연스럽게 암호화통신을 할 수 있도록 해주는 기능입니다.

<그림 6-3>은 아파치 서버에서 Redirect 지시자를 써서 http://test.co.kr 또는 http://www.test.co.kr로 들어온 사용자를 강제로 https://www.test.co.kr로 리다이렉션시켜서 암호화 통신하는 예입니다.

```
<VirtualHost test.co.kr:80>
        ServerAdmin zmnkh@test.co.kr
        ServerName test.co.kr
        ServerAlias www.test.co.kr
        DocumentRoot /home/manpage
        CustomLog logs/test.co.kr-access_log common
        Redirect / https://www.test.co.kr/
</VirtualHost>
```

<그림 6-3> 아파치 서버에서의 Redirection

또다른 방법으로는 OS나 Web Programming 언어의 종류에 상관없이 모두 공통적으로 사용하는 HTML tag를 이용한 방법으로써, 어떤 경우에서나 적용이 가능하기 때문에 가장 많이 이용되고 있습니다.

<그림 6-4>은 웹페이지의 index.html에 한 줄의 소스코드를 추가함으로써 http://URL로 접속하는 사용자들을 강제로 https://URL로 리다이렉션하는 예입니다.

```
<meta http-equiv='refresh' content='0; url=https://[redacted].co.kr/index.html' target='_top'>
```

<그림 6-4> HTML Tag를 이용한 Redirection

위와 같이 Meta 태그를 이용하는 경우, 1초 정도 깜빡하는 현상이 나타나기 때문에 종종 Javascript를 이용하기도 합니다.

Meta tag를 이용한 html Redirection 방법과 동일하게, 사용자들이 익숙하게 접속하는 http://www.test.com의 index 페이지에 삽입해 두면, 사용자들이 불편하게 https://라는 프로토콜을 특별히 지정해 주지 않아도, 보안을 위해서 암호화 통신이 적용된 https:// www.test.com으로 리다이렉션해주게 됩니다.

```
<script>
var url = "https://www.test.com";
window.location.replace(url);
</script>
```

<그림 6-5> Javascript를 이용한 Redirection

나. 페이지별 암호화하기

페이지별 암호화는 현재 위치하고 있는 페이지에서 다른 페이지로 이동할 때, 보안을 위해서 암호화된 전송을 할 것인지 아니면 평문 전송할 것인지를 선택하여 암호화하는 것을 말합니다.

부분적인 페이지 암호화를 사용하는 이유는 암호화 적용이 필요없는 부분까지 암호화를 하여 서버의 부하를 증가시키는 것을 최대한 줄일 수 있기 때문입니다.

다음 <그림 6-6>은 사이트의 메뉴 부분 예입니다. 이 중 '서버관련 강좌 & TIP' 메뉴를 클릭하여 이동을 하면 https가 호출되어 서버와 클라이언트간의 통신이 암호화되어 전송되고, 'Q&A' 메뉴를 클릭하여 이동하면 http가 호출되어 서버와 클라이언트간의 통신이 평문으로 이루어지게 하는 방법을 알아보겠습니다.

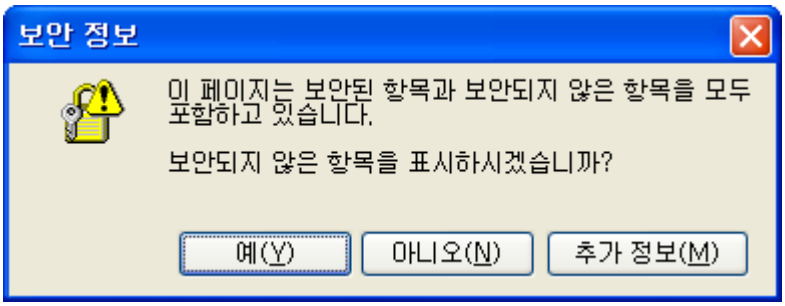
<그림 6-6> 페이지별 암호화 대상 메뉴

<그림 6-7>은 위 메뉴 부분의 소스코드입니다. 빨간색 밑줄 부분 중 첫 번째 밑줄에 해당하는 부분이 현재 위치에서 메뉴를 클릭하여 이동을 때 암호화 전송을 하도록 하게끔 설정된 것이고, 두 번째 밑줄은 현재 위치에서 메뉴를 클릭하여 이동할 때 평문 전송을 하도록 설정된 것입니다.

```
<map name="ImageMap1">
<area shape="rect" coords="193, 74, 249, 90" href="onlinebook/online.ht
m" target="main">
<area shape="rect" coords="267, 75, 401, 89" href="https://[illegible].co.k
r/zboard/zboard.php?id=lecture" target="_top">
<area shape="rect" coords="423, 73, 479, 89" href="https://[illegible].co.k
r/zboard/zboard.php?id=problem" target="_top">
<area shape="rect" coords="497, 73, 537, 89" href="http://[illegible].co.kr
/zboard/zboard.php?id=qna" target="_top">
<area shape="rect" coords="555, 73, 609, 89" href="http://[illegible].co.kr
/zboard/zboard.php?id=down" target="_top">
<area shape="rect" coords="679, 5, 717, 23" href="index.html" target="_
top">
```

<그림 6-7> 페이지별 암호화 대상 메뉴의 소스코드

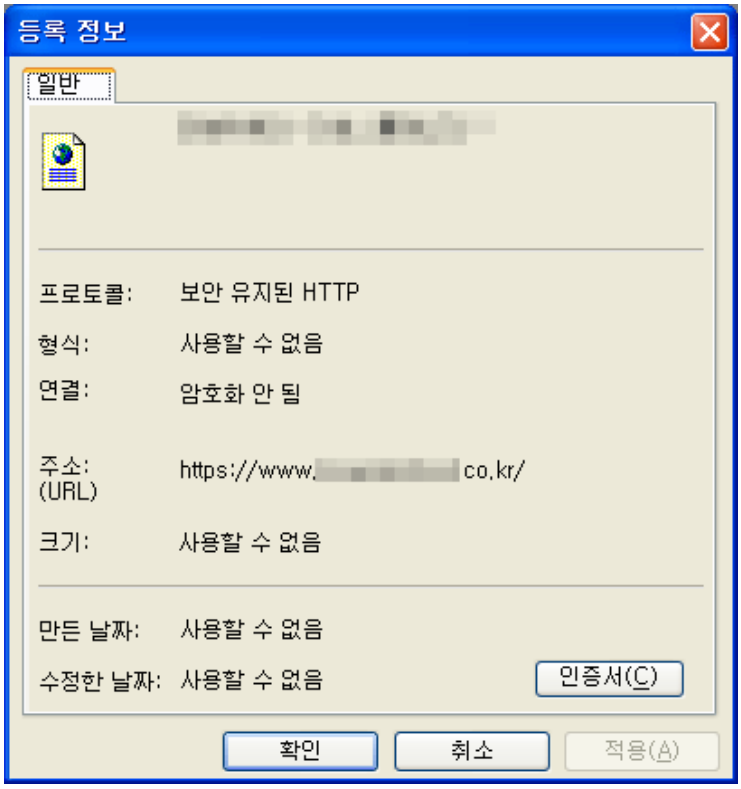
이렇게 페이지별로 암호화가 적용된 사이트를 방문해보면, <그림 6-8>과 같은 경고창을 만나게 되는 경우가 있습니다.



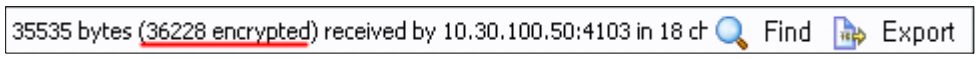
<그림 6-8> SSL이 적용된 페이지의 경고창

이 경고창이 뜨는 것는 암호화 통신을 유지하기 위해서는 웹페이지내의 모든 URL의 호출이 https://로 이루어져야 하나, http:// 즉 평문 통신을 위한 웹페이지 URL이 포함되어 있다는 것을 의미합니다.

이런 경고창이 발생하는 웹 페이지 속성을 보면 <그림 6-9>처럼 “암호화 안됨”이라고 해서 마치 암화화가 되지 않은 평문 상태로 데이터가 전송되어지는 것처럼 생각되지만 웹 페이지간 전송되는 데이터를 볼 수 있는 third-parth 툴을 이용해서 확인해 보면, <그림 6-10>와 같이 암호화 통신이 이루어지고 있다는 것을 알 수가 있습니다.

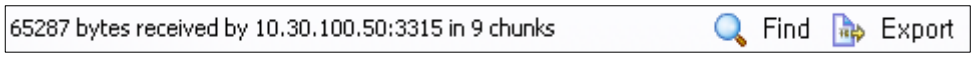


<그림 6-9> http 평문 통신 주소가 호출되는 웹페이지의 속성



<그림 6-10> https를 통한 암호화 통신

<그림 6-11>는 <그림 6-10>의 결과와 비교하기 위해서 암호화 되지 않은 평문 통신(http) 상태를 나타낸 그림입니다.



<그림 6-11> http를 통한 평문 통신

하지만 <그림 6-8>와 같이 경고창이 발생하게 되면, 상세한 내용을 모르고 웹사이트에 접속하는 사용자들에게 보안이 되고 있지 않다는 불신을 줄 수도 있고, 또한 계속적인 경고창으로 인해서 불편해 할 수 있으므로 가급적 발생하지 않도록 웹 페이지내의 모든 URL을 https://로 바꿔주는 것이 좋습니다.

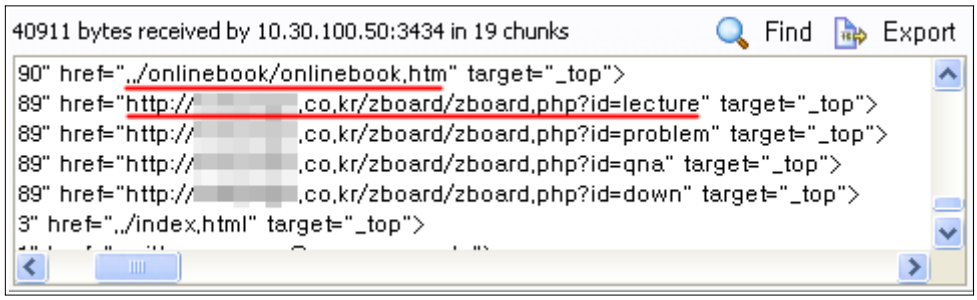
만일 절대경로로 호출하는 것이 아니라, 상대경로로 호출하는 것이라면 소스를 변경하지 않아도 됩니다.

※ 참고 : 절대경로와 상대경로

절대경로 호출과 상대경로 호출이란 무엇인가?

절대경로란 내가 열어보고자 혹은 내가 가고자 하는 웹페이지의 경로를 전체적으로 기술하는 것이고, 상대경로란 내가 현재 있는 위치를 기준으로 내가 열어보고자 혹은 내가 가고자 하는 웹페이지의 경로를 기술하는 것을 말한다.

아래 그림에서 첫 번째 밑줄 그은 부분이 상대경로로 호출하는 경우이고, 두 번째 밑줄 그은 부분이 절대경로로 호출하는 경우이다.
첫 번째의 경우에는 https 암호화 통신을 하더라도 소스코드 수정이 필요없는 부분이고, 두 번째의 경우에는 https 암호화 통신을 할 경우 호출 URL을 http에서 https로 바꿔줘야 한다. 만일 바꿔주지 않을 경우에는 <그림 6-8>과 같이 경고창이 뜨게 된다.



다. 프레임별 암호화하기

SSL을 이용한 보안포트(443)를 웹페이지에 적용하는 방법을 앞서 소개하였습니다. 단순히 http를 https로만 바꾸어주면 보안포트를 이용해서 암호화 통신을 할 수 있었습니다.

하지만, 프레임이 삽입된 웹페이지의 경우에는 약간 적용하는 방식이 다르기 때문에 소개하고자 합니다. 프레임이 적용된 페이지를 이용하면 암호화된 페이지와 비 암호화된 페이지를 각각 적용시킬 수 있습니다.

적용 시나리오는 <그림 6-12>과 같이 웹페이지(index.html)에 프레임으로 두 개의 페이지 topmenu.htm과 main.htm을 불러오는 소스코드가 있을 때 소스코드의 URL을 <그림 6-13>와 <그림 6-14>처럼 변경하고 웹 브라우저에서 http와 https로 각각 호출했을 때의 결과를 살펴보고자 합니다.

```
<html>

<head>
<meta http-equiv="content-type" content="text/html; charset=euc-kr">
<title>SSL Frame Test</title>
</head>
<frameset rows="100, 1*" border="1">
   <frame src="http://lab.[redacted].co.kr/test_ssl/topmenu.htm" scrol
ling="yes" name="top" namo_target_frame="main">
    <frame src="http://lab.[redacted].co.kr/test_ssl/main.htm" scrolli
ng="yes" name="main">
    <noframes>
    <body bgcolor="white" text="black" link="blue" vlink="purple" ali
nk="red">
    <p>SSL Frame Test의 페이지 입니다. <br> 이페이지를 보기 위해서는
       프레임을 볼수 있는 웹 브라우저가 필요합니다.</p>
    </body>
    </noframes>
</frameset>

</html>
```

<그림 6-12> 프레임이 포함된 웹페이지

```
<html>

<head>
<meta http-equiv="content-type" content="text/html; charset=euc-kr">
<title>SSL Frame Test</title>
</head>
<frameset rows="100, 1*" border="1">
   <frame src="https://lab.[redacted].co.kr/test_ssl/topmenu.htm" scro
lling="yes" name="top" namo_target_frame="main">
    <frame src="http://lab.[redacted].co.kr/test_ssl/main.htm" scrolli
ng="yes" name="main">
    <noframes>
    <body bgcolor="white" text="black" link="blue" vlink="purple" ali
nk="red">
    <p>SSL Frame Test의 페이지 입니다. <br> 이페이지를 보기 위해서는
       프레임을 볼수 있는 웹 브라우저가 필요합니다.</p>
    </body>
    </noframes>
</frameset>

</html>
```

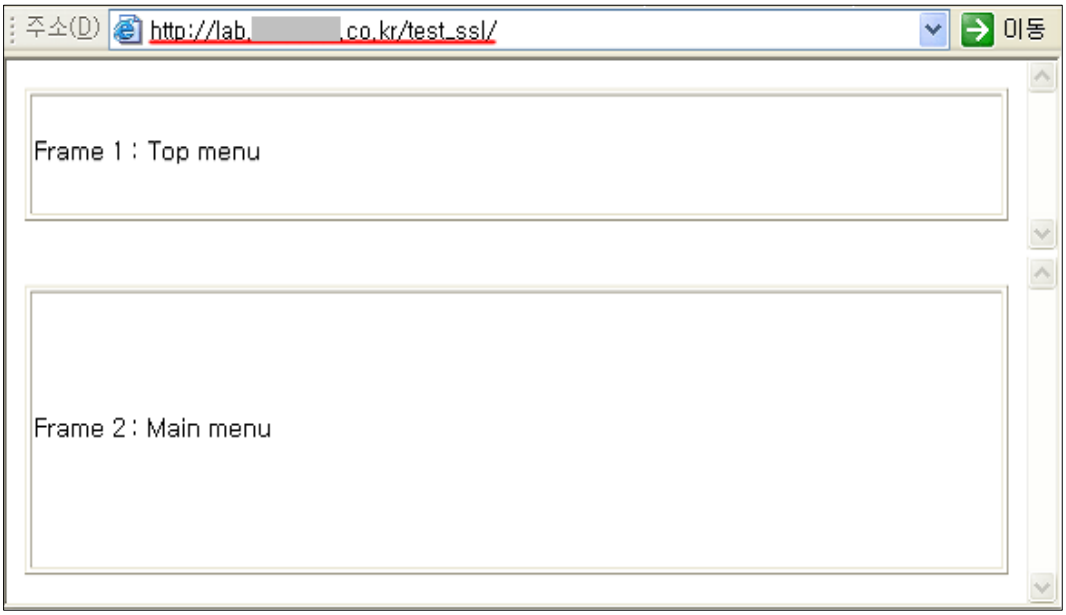
<그림 6-13> topmenu.htm을 https로 호출하기

```
<html>

<head>
<meta http-equiv="content-type" content="text/html; charset=euc-kr">
<title>SSL Frame Test</title>
</head>
<frameset rows="100, 1*" border="1">
   <frame src="https://lab.[redacted].co.kr/test_ssl/topmenu.htm" scro
lling="yes" name="top" namo_target_frame="main">
    <frame src="https://lab.[redacted].co.kr/test_ssl/main.htm" scroll
ing="yes" name="main">
    <noframes>
    <body bgcolor="white" text="black" link="blue" vlink="purple" ali
nk="red">
    <p>SSL Frame Test의 페이지 입니다. <br> 이페이지를 보기 위해서는
       프레임을 볼수 있는 웹 브라우저가 필요합니다.</p>
    </body>
    </noframes>
</frameset>

</html>
```

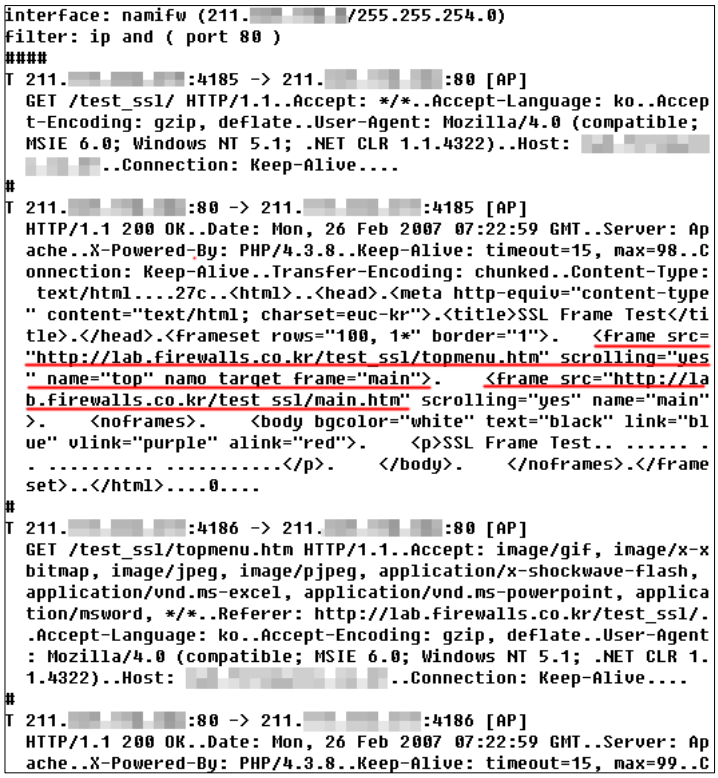
<그림 6-14> topmenu.htm과 main.htm을 https로 호출하기

① 비암호화 통신(http)를 이용해서 호출하기

<그림 6-15>는 topmenu.htm과 main.htm을 모두 <그림 6-12>의 소스를 이용해서 호출한 경우입니다. 이 경우에는 모든 정보가 암호화되지 않고 <그림 6-16>와 같이 그대로 노출됩니다.

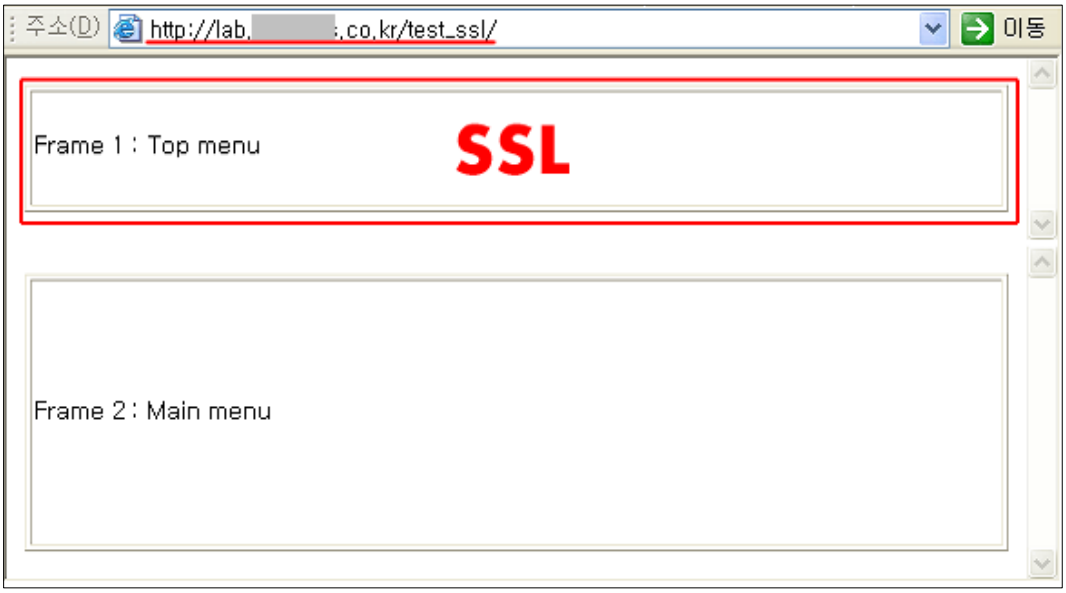


<그림 6-15> 비암호화된 페이지 호출하기



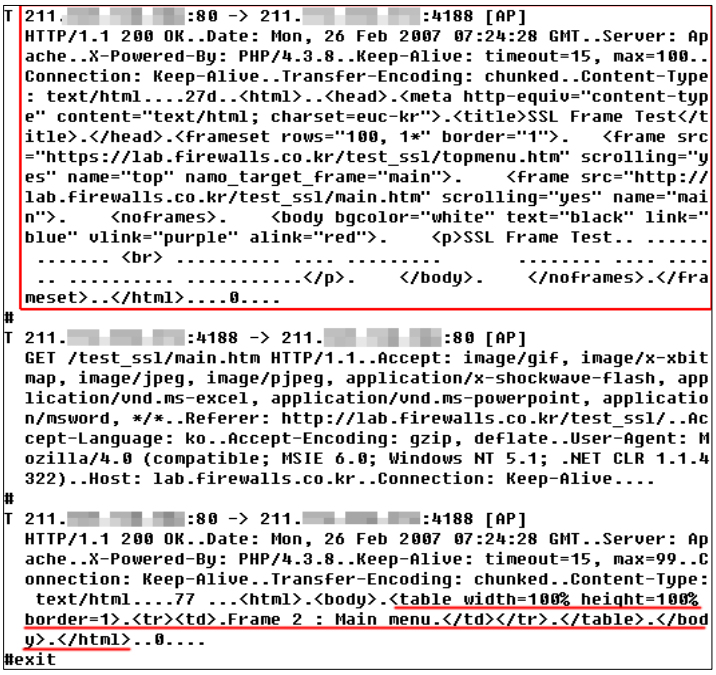
<그림 6-16> HTTP 호출시 80 포트 모니터링 결과

다음으로는 <그림 6-13>의 소스코드를 적용하여 topmenu.htm만을 https로 호출을 하는 경우입니다.



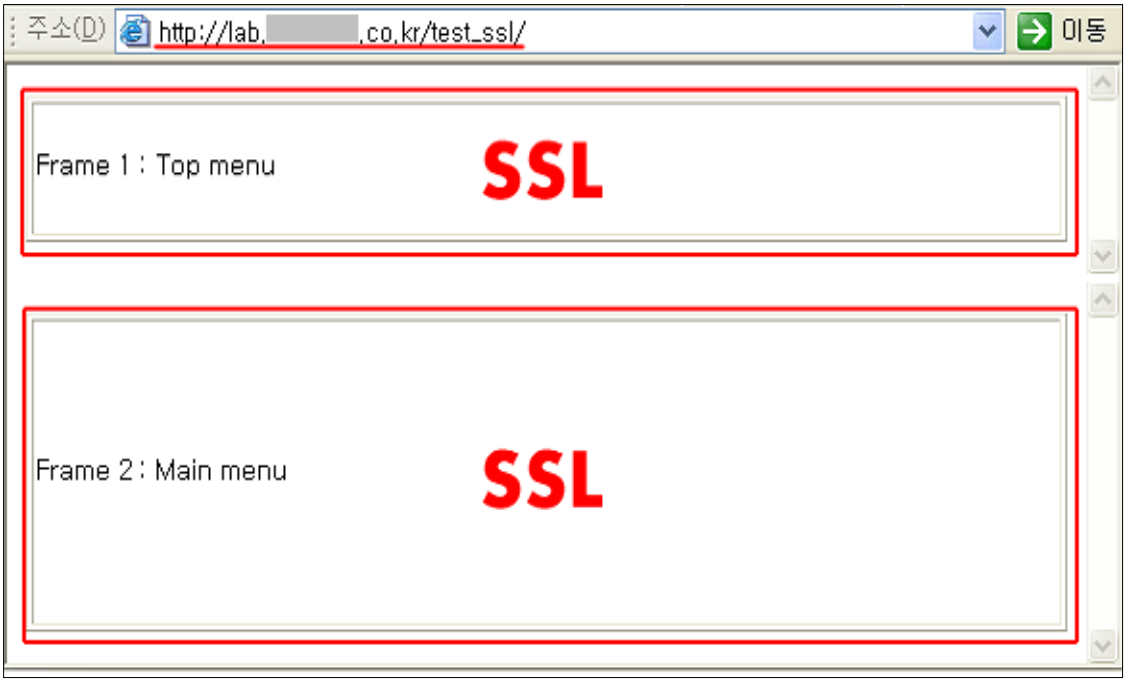
<그림 6-17> topmenu.htm만 암호화하여 호출하기

프레임을 이용해서 호출하는 경우에는 아래 <그림 6-18>과 같이 암호화되지 않는 index.html (빨간색 네모박스)의 내용과 main.htm의 내용만이 80 포트로 텍스트 전송되는 것을 알 수 있습니다. topmenu.htm의 내용은 암호화 전송되어지기 때문에 평문 전송되는 80 포트에서는 내용이 확인되지 않습니다.



<그림 6-18> topmenu.htm의 내용만 암호화된 모니터링 결과

마지막으로 <그림 6-14>과 같이, 호출하는 index.html을 제외하고 모든 프레임내의 호출페이지를 https를 통해서 호출하게 될 경우에는 아래와 같이 index.html의 내용만 평문으로 전송이 되고, 나머지 topmenu.htm과 main.htm은 암호화 되어서 전송됩니다.



<그림 6-19> topmenu.htm과 main.htm을 https로 호출하기

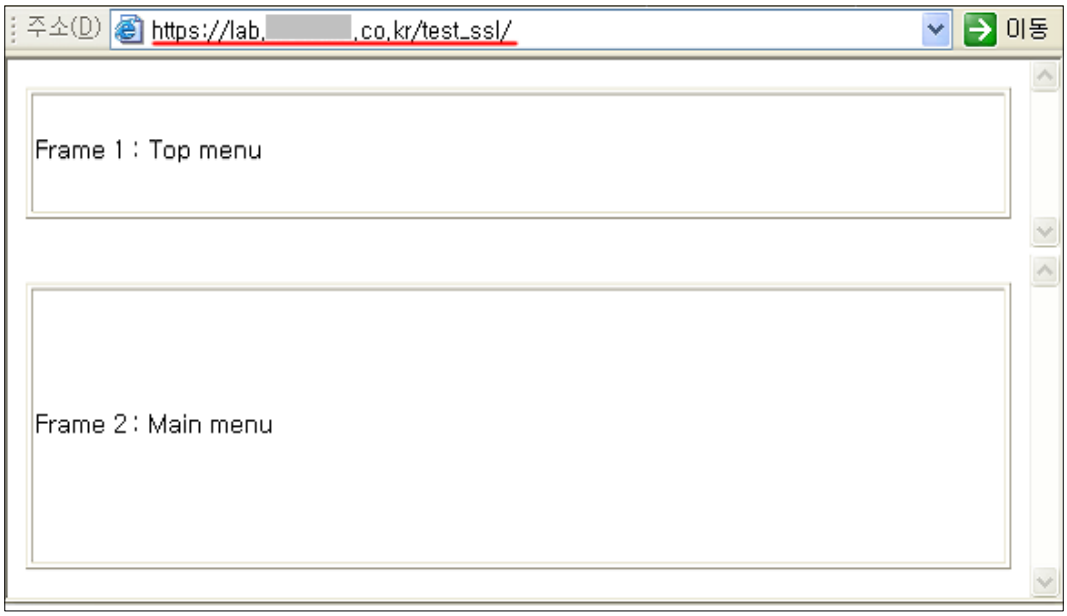
```
interface: namifw (211.[redacted]/255.255.254.0)
filter: ip and ( port 80 )
########
T 211.[redacted]:4190 -> 211.[redacted]:80 [AP]
  GET /test_ssl/ HTTP/1.1..Accept: */*..Accept-Language: ko..Accep
  t-Encoding: gzip, deflate..User-Agent: Mozilla/4.0 (compatible;
  MSIE 6.0; Windows NT 5.1; .NET CLR 1.1.4322)..Host: lab.firewall
  s.co.kr..Connection: Keep-Alive....
##
T 211.[redacted]:80 -> 211.[redacted]:4190 [AP]
  HTTP/1.1 200 OK..Date: Mon, 26 Feb 2007 07:26:05 GMT..Server: Ap
  ache..X-Powered-By: PHP/4.3.8..Keep-Alive: timeout=15, max=100..
  Connection: Keep-Alive..Transfer-Encoding: chunked..Content-Type
  : text/html....27e..<html>..<head>.<meta http-equiv="content-typ
  e" content="text/html; charset=euc-kr">.<title>SSL Frame Test</t
  itle>.</head>.<frameset rows="100, 1*" border="1">.   <frame src
  ="https://lab.firewalls.co.kr/test_ssl/topmenu.htm" scrolling="y
  es" name="top" namo_target_frame="main">.    <frame src="https:/
  /lab.firewalls.co.kr/test_ssl/main.htm" scrolling="yes" name="ma
  in">.    <noframes>.    <body bgcolor="white" text="black" link=
  "blue" vlink="purple" alink="red">.    <p>SSL Frame Test.. .....
  . ....... <br> .......... .... .........       ........ .... ...
  . .. .......... ..........</p>.    </body>.    </noframes>.</fr
  ameset>..</html>....0....
####exit
```

<그림 6-20> index.html의 내용만이 모니터링된 결과

② 암호화 통신(https)을 이용해서 호출하기

앞에서와 같은 절차를 이용해서 https를 이용해서 <그림 6-11>과 같이 호출을 하게 되면, 프레임을 포함하고 있는 index.html은 URL을 https로 호출을 하게 되므로, 항상 암호화가 되어지고, topmenu.htm과 main.htm은 <그림 6-12>,<그림 6-13>,<그림 6-14>과 같이 암호화

적용 여부에 따라서, 평문 통신 또는 암호화 통신이 이루어집니다.



<그림 6-21> https를 이용한 호출

```
interface: namifw (211.                254.0)
filter: ip and ( port 80 )
#####
T 211.          :4183 -> 211.          :80 [AP]
  GET /test_ssl/topmenu.htm HTTP/1.1..Accept: image/gif, image/x-x
  bitmap, image/jpeg, image/pjpeg, application/x-shockwave-flash,
  application/vnd.ms-excel, application/vnd.ms-powerpoint, applica
  tion/msword, */*..Accept-Language: ko..Accept-Encoding: gzip, de
  flate..User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT
   5.1; .NET CLR 1.1.4322)..Host:                ..Connection:
   Keep-Alive....
##
T 211.          :80 -> 211.          :4183 [AP]
  HTTP/1.1 200 OK..Date: Mon, 26 Feb 2007 07:21:34 GMT..Server: Ap
  ache..X-Powered-By: PHP/4.3.8..Keep-Alive: timeout=15, max=100..
  Connection: Keep-Alive..Transfer-Encoding: chunked..Content-Type
  : text/html....76 ..<html>.<body>.<table width=100% height=100%
  border=1>.<tr><td>.Frame 1 : Top menu.</td></tr>.</table>.</body
  >.</html>...0....
####
T 211.          :4184 -> 211.          :80 [AP]
  GET /test_ssl/main.htm HTTP/1.1..Accept: image/gif, image/x-xbit
  map, image/jpeg, image/pjpeg, application/x-shockwave-flash, app
  lication/vnd.ms-excel, application/vnd.ms-powerpoint, applicatio
  n/msword, */*..Accept-Language: ko..Accept-Encoding: gzip, defla
  te..User-Agent: Mozilla/4.0 (compatible; MSIE 6.0; Windows NT 5.
  1; .NET CLR 1.1.4322)..Host:                ..Connection: Ke
  ep-Alive....
##
T 211.          :80 -> 211.          :4184 [AP]
  HTTP/1.1 200 OK..Date: Mon, 26 Feb 2007 07:21:35 GMT..Server: Ap
  ache..X-Powered-By: PHP/4.3.8..Keep-Alive: timeout=15, max=100..
  Connection: Keep-Alive..Transfer-Encoding: chunked..Content-Type
  : text/html....77 ...<html>.<body>.<table width=100% height=100%
   border=1>.<tr><td>.Frame 2 : Main menu.</td></tr>.</table>.</bo
  dy>.</html>..0....
##exit
```

<그림 6-22> https 호출시 80 포트 모니터링 결과

<그림 6-22>을 보면, 웹 브라우저에서 https를 통해서 호출한 index.html의 내용은 암호화되어서 통신이 이루어지기 때문에 80 포트를 모니터링 하였을 경우에 그 내용이 보이지 않지만, index.html 안에 있는 프레임을 통해서 http로 호출한 topmenu.htm과 main.htm은 https 통신을 통해서 index.html을 호출했지만, 평문으로 통신이 되는 것을 확인할 수 있습니다.

<그림 6-13>, <그림 6-14>의 소스를 같은 방법으로 테스트 해보면, http로 호출된 웹페이지는 암호화 통신이 이루어지지 않고 있는 것을 알 수 있습니다.

이와 같이 프레임을 이용하면, 필요에 따라서 한 페이지에서 암호화가 제공되는 부분과 암호화가 제공되지 않는 부분이 공존할 수 있도록 구성할 수 있지만, 앞서서도 이미 언급했듯이 아무리 웹 브라우저에서 https를 이용해서 호출을 했어도 프레임으로 불러오는 페이지가 http 주소를 가지고 있을 경우에는 암호화가 되지 않고 정보의 노출이 발생할 수 있으므로, 프레임이 사용되는 페이지를 암호화를 위해서 https로 호출하고자 할 때에는 꼭 확인을 해보시기 바랍니다.

라. 체크박스를 이용한 선별적 암호화하기

웹페이지 전체를 암호화하지 않고 선별적으로 암호화하는 경우, 정보입력시 보안접속을 체크함으로써 프로토콜을 호출하는 방법이 있습니다. 다음은 로그인 박스에서 선별적으로 암호화된 통신을 하기 위한 HTML 소스코드의 예입니다.

* 소스 코드	* 소스 코드

<script language="JavaScript">
< !--
function checkLoginForm1() {
var f = document.forms["LoginForm1"];
//아이디 입력 검사
if(f.memberID.value==""){
alert("아이디를 입력하세요");
f.memberID.focus();
return false;
}
//비밀번호 입력 검사
if(f.memberPW.value==""){
alert("비밀번호를 입력하세요");
f.memberPW.focus();
return false;
}
//액션
f.action = "h
http://login.your-domain.com/login1.html;
 	
return true;
}
//-->
< /script>
< form name="LoginForm1" method="POST"
onSubmit="return checkLoginForm1();">
< table>
<tr>
<td>아이디</td>
<td><input type="text"
name="memberID"></td>
<td> </td>
 	
</tr>
<tr>
<td>비밀번호</td>
<td><input type="password"
name="memberPW"></td>
	<script language="JavaScript">
	<!--
	function checkLoginForm2() {
	var f = document.forms["LoginForm2"];
	//아이디 입력 검사
	if(f.memberID.value==""){
	alert("아이디를 입력하세요");
	f.memberID.focus();
	return false;
	}
	//비밀번호 입력 검사
	if(f.memberPW.value==""){
	alert("비밀번호를 입력하세요");
	f.memberPW.focus();
	return false;
	}
	//액션
	if (f.SSL_Login.checked) { //보안접속 체크
	판별
	//보안접속을 체크했을 때의 액션
	f.action =
	"https://login.your-domain.com/login1.html";
	} else {
	//보안접속을 체크하지 않았을 때의 액션
	f.action =
	"http://login.your-domain.com/login1.html";
	}
	return true;
	}
	//-->
	</script>
	<form name="LoginForm2" method="POST"
	onSubmit="return checkLoginForm2();">
	<table>
	<tr>
	<td>아이디</td>
	<td><input type="text"
	name="memberID"></td>
	<td><input type="checkbox" value=1
	checkedname="SSL_Login" >보안접속</td>
	</tr>
	<tr>
	<td>비밀번호</td>
	<td><input type="password"
	name="memberPW"></td>

<td><input type="submit" name="Submit" value=" 로그인 "></td> </tr> < /table> < /form>	<td><input type="submit" name="Submit" value=" 로그인 "></td> </tr> </table> </form> ※ SSL을 이용한 암호화된 폼 전송을 하려면, action URL에서 'http' 대신 'https'를 적어 주면 됩니다.

<그림 6-23> 로그인시 보안접속 체크박스를 이용하기 위한 HTML 소스코드

2. 보안서버 적용 확인하기

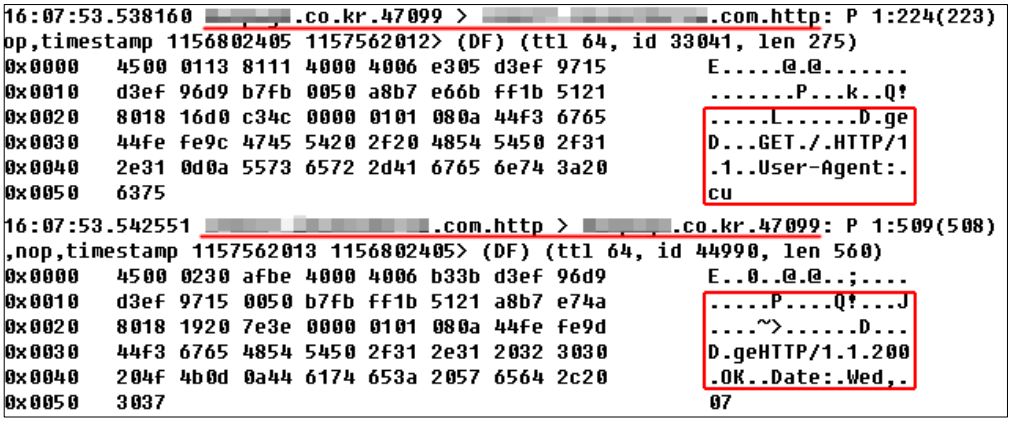
가. 보안서버 적용 확인 방법

앞에서 암호화 통신을 위해서 보안 프로토콜을 적용하는 방법을 알아보았습니다. 본 절에서는 앞서 적용한 보안 프로토콜이 제대로 적용이 되었는지 확인하는 방법에 대해서 알아보겠습니다.

① 패킷 캡쳐를 통한 확인

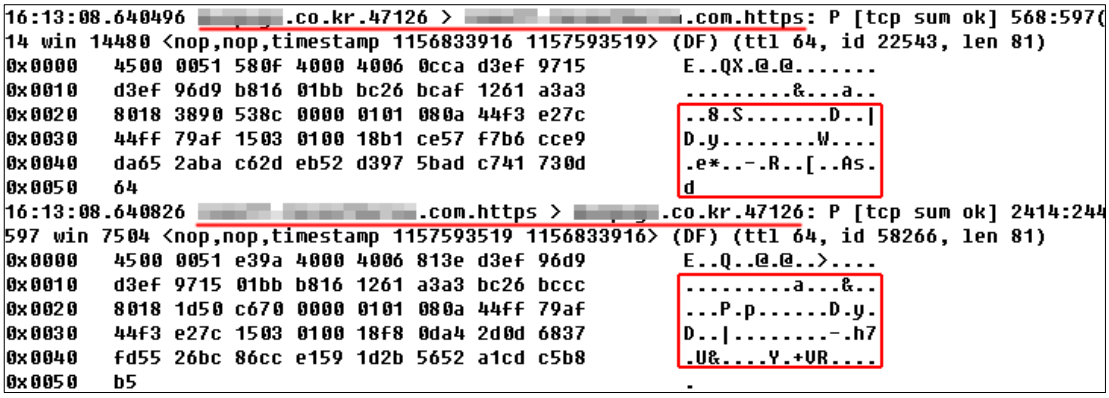
일단 제대로 암호화가 되어지고 있는지 패킷을 캡쳐하여 확인하는 방법입니다.

<그림 6-24>는 일반적인 http를 통한 평문 통신의 경우를 캡쳐한 것입니다. 빨간 네모상자를 확인하면 header의 내용이 평문으로 보이는 것을 확인할 수 있습니다.



<그림 6-24> 평문 통신

<그림 6-25>을 보면 https에 접속 요청을 한 것을 확인할 수가 있으며, 빨간 네모상자를 확인하면 동일구간이지만 header의 내용이 암호화 되어 내용을 알아볼 수 없게 되었음을 알 수 있습니다. 즉, 암호화 통신이 정상적으로 이루어진다는 것을 의미합니다.



<그림 6-25> 암호화된 통신

② 웹페이지에서 확인

직접 패킷을 캡쳐해서 확인하는 방법 외에도, 웹페이지에서 간단히 암호화가 되어지고 있는지를 확인하는 방법이 있습니다.

SSL이 적용된 웹사이트에 https 프로토콜로 접속을 했을 경우에, <그림 6-26>과 같이 브라우저 하단에 자물쇠 모양의 표시가 나타나는 것을

확인할 수 있을 것입니다. 현재 사이트와의 통신이 암호화되어서 진행되고 있다는 것을 보여주는 것입니다. 웹사이트 구성방법에 따라 자물쇠 이미지가 보이지 않을 수 있으며, 구축 방법에 따라 모양은 다르게 나타날 수 있습니다.

<그림 6-26> 암호화 통신이 이루어지고 있음을 보여주는 자물쇠 이미지

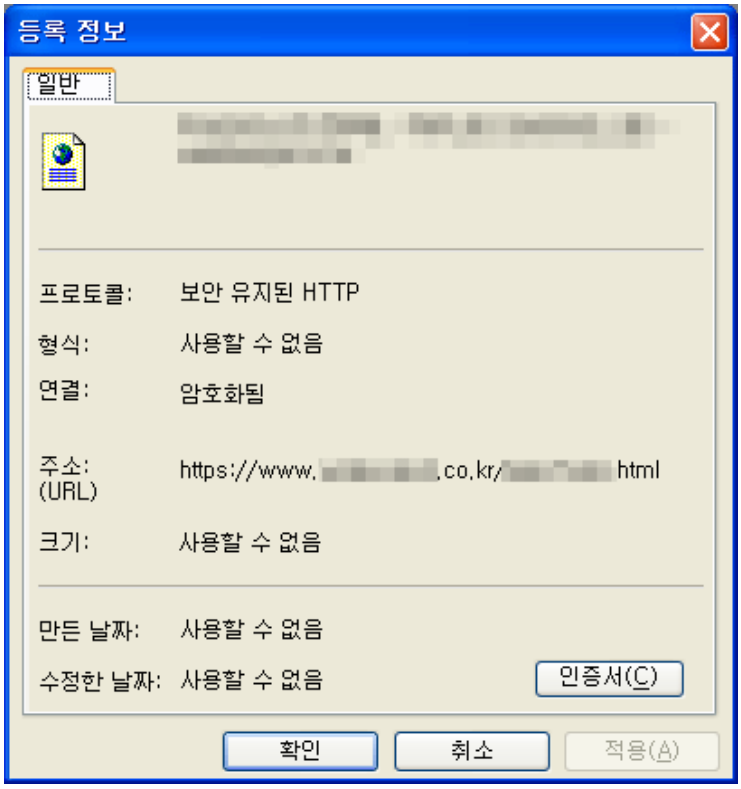
③ 호출시 포트번호 확인

https를 이용하여 접속하시면 일반적으로 443번 포트로 연결이 되어 SSL 인증서버를 통한 통신을 하게 됩니다.

서버에 여러 개의 인증서버를 설치할 경우 상황에 따라 443번 포트가 아닌 여러 가지 포트를 이용해서 접속을 하는 상황이 발생할 수 있습니다. 이런 경우 설치를 대행하는 업체나 호스팅업체에서 임의의 포트를 지정하거나 사용할 포트를 지정받아 SSL 인증서를 설치한 뒤 작업완료 통보와 함께 사용된 포트번호를 알려주게 됩니다.

④ 웹페이지 속성보기를 통한 확인

암호화를 적용한 웹페이지가 정상적으로 암호화되고 있는지는 웹페이지에서 오른쪽 마우스를 클릭하고 속성을 선택한 후 웹페이지 등록 정보를 통하여 확인할 수가 있습니다. 현재 페이지가 보안이 되고 있다면 <그림 6-27>와 같은 웹페이지 속성을 확인할 수 있습니다.



<그림 6-27> 보안이 적용된 웹페이지 등록정보

나. 인증서의 암호화 상태 확인 방법

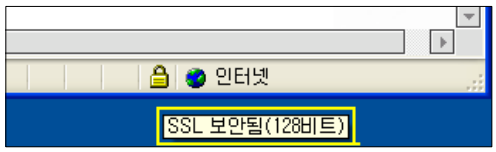
접속한 웹페이지가 암호화되고 있다는 것을 확인한 후, 다음과 같은 단계를 걸쳐서 설치된 인증서의 암호화 상태를 확인할 수 있습니다.

① 웹페이지 접속

인증서가 설치된 웹페이지에 접속하십시오.

<그림 6-28> 보안이 적용된 웹페이지 접속

브라우저 하단에 있는 자물쇠 아이콘에 마우스를 올리면 암호화 방식에 대한 확인이 가능합니다.

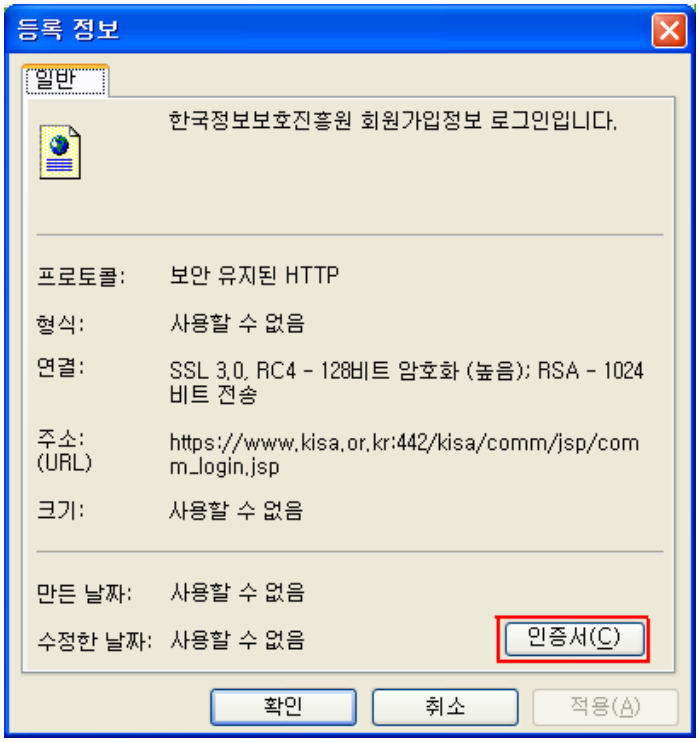


<그림 6-29> 자물쇠 이미지를 통한 암호화 방식 확인

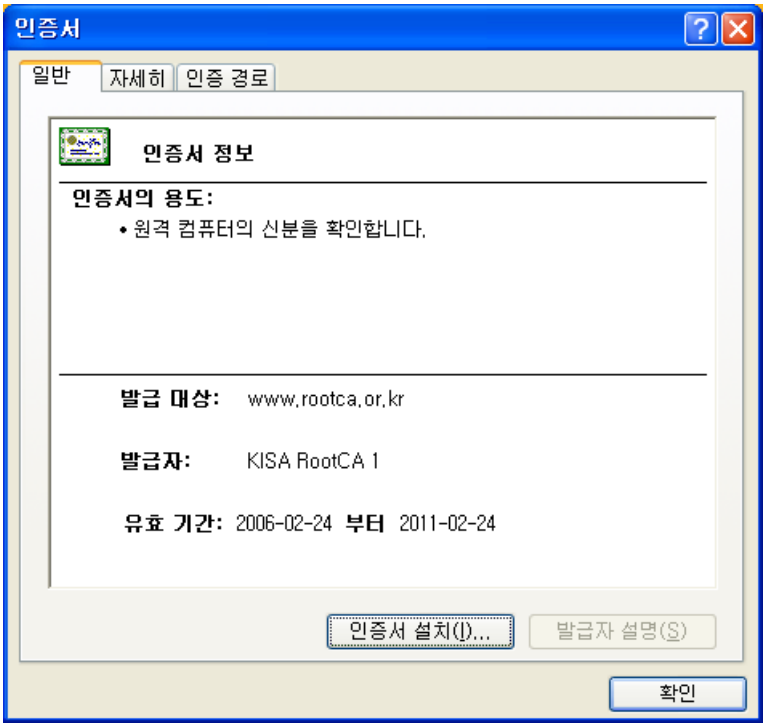
② 인증서 기본 정보 확인

설치된 인증서의 정보를 확인하려면 자물쇠 아이콘을 더블클릭하거나 브라우저에서 마우스 오른쪽 버튼을 클릭하시면 등록정보 창이 열리게 됩니다.

등록정보 창 하단에 인증서 버튼을 선택하여 인증서 창을 열면 설치된 인증서에 대한 정보를 확인할 수 있습니다.



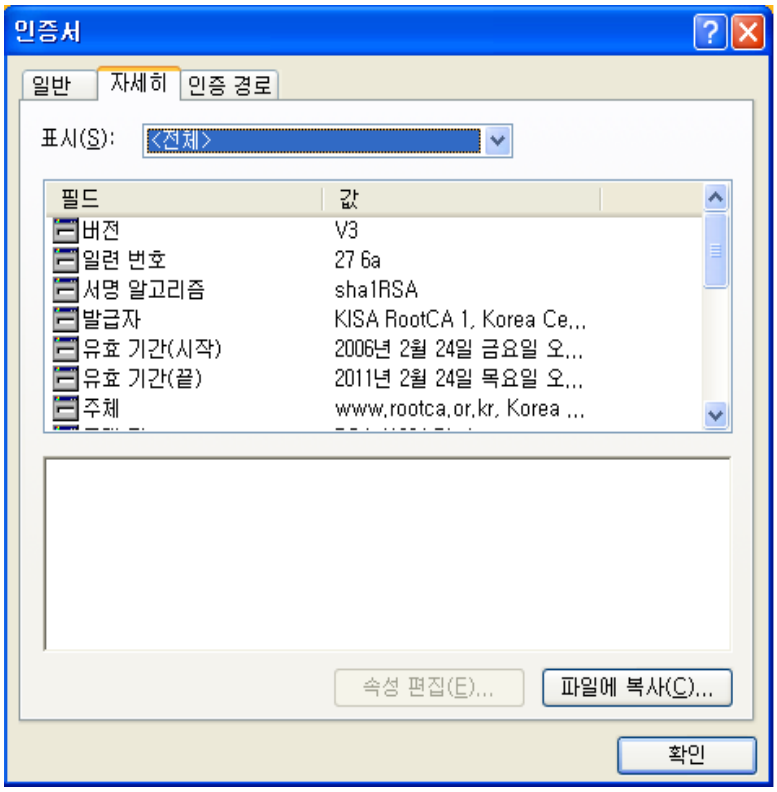
<그림 6-30> 보안이 적용된 웹페이지의 등록정보 중 인증서 버튼



<그림 6-31> 보안이 적용된 웹페이지의 인증서 기본 정보 확인

③ 인증서 상세 정보 확인

발급 대상, 발급자 정보, 발급된 인증서의 유효기간 등 기본적인 인증서 정보를 확인하실 수 있고, 보다 자세한 인증서 관련 정보를 확인하고자 한다면 '자세히' 탭을 선택하면 됩니다.



<그림 6-32> 보안이 적용된 웹페이지의 인증서 상세정보 확인

Ⅶ. 제도 관련 FAQ

① 한국정보보호진흥원(KISA)은 어떤 기관인가요?

정보통신부 산하기관으로 정보통신망법 법률에 의해 인터넷상의 개인정보보호 업무 등을 맡고 있으며, 웹사이트 회원가입 등을 통한 개인정보 수집에 따른 실태조사 및 미흡한 사업자에 대한 개선권고 업무 등을 하고 있습니다.

② 보안서버는 무엇이고, 구축하지 않으면 어떻게 되나요?

보안서버란, PC 이용자들이 웹사이트에 로그인할 때 전송되는 개인정보를 암호화하여 전송하는 기능을 제공하는 암호화 기능이 탑재된 서버를 의미합니다. 귀사의 사이트를 보안서버로 구축하려면 기존의 웹사이트에 개인정보를 암호화해주는 별도의 보안시스템을 구축해 주시면 됩니다.

"정보통신망 이용촉진 및 정보보호 등에 관한 법률"에 의해 보안서버가 구축되지 않은 사이트에 대해서는 1천만원 이하의 과태료 등 행정조치가 있을 수 있습니다.

③ 보안서버 구축이 의무인가요? 관련 법조항이 뭔가요?

보안서버 구축은 현행법상 개인정보를 취급하시는 영리 목적의 사업자 분들은 필수적으로 구축하셔야 하는 의무조항입니다.

관련 법조항은 이미 2005년부터 시행 중이며 "정보통신망 이용촉진 및 정보보호 등에 관한 법률" 제28조(개인정보의 보호조치)와 제67조 (과태료) 규정 등에 명시되어 있습니다. 실제 법조항은 본 가이드의 I장 내용 중 보안서버 관련 규정이나 보안서버전문협의회 홈페이지(www.kisia.or.kr/secureserver)를

참조하시고, 전체 법조항이 필요하신 경우는 정보통신부 또는 법제처 홈페이지를 참조하시기 바랍니다.

④ 보안서버를 꼭 설치해야 하나요? 판매목적의 안내가 아닙니까?

아닙니다. 본 계도활동은 정부통신부의 지도아래 진행되는 사항으로 특정업체를 위한 제품판매나 마케팅 활동이 아닙니다.

⑤ 보안서버 구축하려면 누구에게 연락해야 하나요?

보안서버 구축 전문업체와 상의하시면 되며 'Ⅱ장의 2절 보안서버 구축 전문업체' 내용 또는 보안서버전문협의회 홈페이지 (www.kisia.or.kr/secureserver)를 참조하시기 바랍니다. 이외에 평소에 알고 있던 구축 전문업체를 이용하셔도 됩니다.

⑥ 보안서버 구축하려면 비용이 얼마나 드나요?

구축 방식에 따라 금액의 차이가 크므로 귀사의 사이트에 적합한 보안서버 구축을 위한 자세한 내용은 구축 전문업체와 상의하시기 바랍니다.

⑦ 정확히 언제까지 구축해야 하는 건가요?

관련법규는 2005년부터 이미 시행중이며, 올해 3월부터 모니터링을 통한 사이트 점검 결과에 따라 시정명령과 과태료 부과 등 행정조치가 있을 예정이기 때문에 빠른 시일내에 구축을 완료하셔야 합니다. 구축 계획 일정과 구축결과에 대해 본 사무국(securekorea@kisa.or.kr)으로 알려주시기 바랍니다.

⑧ 지금까지는 규제하지 않다가 왜 이제 와서 이러는 겁니까?

'05년 법개정 후 사업자들의 자율 구축을 유도하였으나 구축이 의무조항이라는 사실을 많이 인지하지 못하셔서 활성화가 되지

못하고 있었습니다. 올해에는 직접 사업자들을 대상으로 홍보 및 계도를 실시한 후 본격적으로 행정조치를 시행할 예정입니다.

⑨ 우리 사이트는 회원가입을 받고 있지 않은데 그래도 구축대상입니까?

회원가입이 아니라도 상담, 주문, 견적, 게시판을 통해 ID/패스워드, 주민번호 등 개인정보를 수집하는 경우도 해당이 됩니다. 다만, 영리목적이 아닌 친목도모를 위한 커뮤니티 등은 해당되지 않습니다.

⑩ 보안서버 구축 대상의 정확한 범위가 어디까지 입니까? 하루에 10명 정도 접속하는 소규모 사이트도 대상이 되는 겁니까?

현행법상의 보안서버 구축대상은 영리목적으로 개인정보를 수집하는 온라인사업자 등입니다. 즉, 일일방문자 수나 사이트 규모에 관계없이 개인정보 수집을 하는 영리목적의 온라인 사업자는 모두 보안서버를 구축하셔야 합니다.
친목도모를 위한 커뮤니티나 비영리 단체의 웹사이트는 해당되지 않습니다. 하지만 이 경우에도 개인정보 수집을 자제하시거나 적절한 보호대책을 적용할 것을 권고하고 있습니다.

⑪ 이미 구축을 완료했는데, 확인도 안하고 계도 메일을 보내는 건 정부기관의 행정처리가 너무 미흡한 것 아닙니까?

수신메일정보는 웹사이트를 통해 확인 가능한 주소로 일괄 발송되었으며, 방대한 수의 온라인 사이트를 대상으로 사업을 진행하다보니 일부 기구축 사이트가 메일이 포함된 경우가 있습니다. 이 경우 본 사무국에 보안서버가 기구축되었음을 신고하여 주시면 됩니다. 그리고, 신규사업운영이나 협력업체 관리 등에 참조하시라는 의미로 받아주시면 감사하겠습니다.

⑫ 만약 더 이상 개인정보를 수집하지 않으면 어떻게 됩니까?

법률적용 대상이 되지 않기 때문에 보안서버 구축이 필요없으며, 운영상 불필요한 개인정보 수집은 하지 않는 것이 바람직합니다.
다만 이 경우, 기존에 수집된 개인정보를 폐기하고 수집을 중단하겠다는 내용을 명시한 이메일을 사무국에 송부해 주시기 바랍니다. (securekorea@kisa.or.kr)

⑬ 사이트를 현재 운영하지 않거나 혹은 조만간 폐쇄할 예정입니다.

"사이트 폐쇄" 또는 "폐쇄 예정"(날짜 기입 요망)임을 명시하는 이메일을 송부 부탁드립니다. (securekorea@kisa.or.kr)

⑭ 피싱(Phishing)이 뭔가요? 보안서버를 구축하면 피싱도 예방되나요?

피싱이란 금융기관 등의 웹사이트나 거기서 보내온 메일로 위장하여 개인의 ID 및 패스워드, 신용카드번호, 계좌정보 등을 빼내 이를 불법적으로 이용하는 사기수법을 의미합니다.
보안서버가 구축된 사이트를 대상으로 피싱 공격을 시도하기 어렵기 때문에 피싱에 의한 피해를 줄이는 효과도 있습니다.

⑮ 구축 업체를 소개해 줄 수는 없습니까?

저희는 특정업체를 소개해드릴 수는 없습니다. 가이드 본문에 포함된 보안서버 구축 전문업체 연락처를 참고하시거나 기타 보안서버 구축업체를 통해 문의해 주시기 바랍니다.

⑯ 보안서버 관련해서 더 자세한 정보는 없나요?

보안서버전문협의회 홈페이지 (www.kisia.or.kr/secureserver)를 참조하시기 바랍니다.

⑰ 저희 사이트는 전체 계열사의 회원정보를 한 곳에서 관리하는 통합로그인 정책에 따라 운영되기 때문에 개인정보를 보유하고 있지 않습니다. 저희도 대상인가요?

> 통합로그인 정책에 따라 운영되는 사이트의 경우, 로그인외에 개인정보의 요구가 있는 경우는 보안서버 구축 대상이 됩니다. 예를 들어, 통합로그인 후 게시판 이용시 개인정보를 요구하는 경우 암호화를 해야 합니다.

⑱ 웹사이트 관리를 호스팅 업체에 맡기고 있는데, 보안서버 구축도 호스팅 업체가 해야 하는 거 아닙니까?

> 보안서버 구축 의무는 통상 사이트를 통해 개인정보를 수집하시는 사이트 담당자에게 귀속되나 구축과 관련된 자세한 사항은 호스팅 업체와 상의하시는 것이 좋습니다.

⑲ 웹 호스팅을 받고 있는데 호스팅업체가 보안서버 지원을 하지 않는다고 합니다. 어떻게 해야 하나요?

> 웹호스팅서비스 제공업체가 지원을 하지 않는다면 정보통신부 또는 한국정보보호진흥원에 문의하시기 바랍니다.

⑳ 암호화되어야 하는 개인정보의 범위는 어디까지입니까?

> 보안서버 구축 등의 조치를 통해 암호화해야 하는 개인정보의 대표적인 예로는 로그인시 ID/패스워드, 회원가입시 주민번호, 인터넷 뱅킹 이용시 계좌 번호/계좌 비밀번호 등이 해당됩니다. 또한 게시판 등 이용시 이름과 전화번호, 메일주소 등 두 가지 이상의 정보가 결합하여 개인의 식별이 가능해지는 경우 또한 암호화하여야 합니다.

※ 추가적인 문의사항은, 아래 이메일로 문의해 주시기 바랍니다.
이메일 : securekorea@kisa.or.kr

부록 A. 멀티도메인 SSL 인증서 소개

1. 배경(Background)

가. 목적

지금까지의 SSL 인증서는 하나의 도메인에 대해서만 인증을 해주고 있기 때문에, 한 대의 서버에서 한 개의 인증서만을 사용하는 것이 일반적이었습니다.

웹 서버 암호화 통신을 위해서 https가 응답하게 되는 기본적인 443 포트를 하나의 도메인만이 소유하고 사용할 수가 있기 때문에, 한 대에 여러 개의 도메인이 존재하고 모든 도메인이 https를 사용하고자 하면, 아래의 두 가지 방법을 사용하였습니다.

첫 번째는 한 대의 서버에 도메인 개수만큼의 각기 다른 IP를 부여하는 것입니다. 이는 IP부족으로 인해서 택하게 가상호스팅 방식(한 서버에서 하나의 IP로 여러 개의 도메인을 운영하는 방식)에 반하게 되는 문제점이 발생하며, IP를 추가적으로 할당할 수 없는 경우가 많기 때문에 잘 사용되지 못하는 경우입니다.

나머지 한 가지는 가장 많이 사용하는 방식으로 기본 https 포트인 443이 아닌, 다른 포트를 사용해야 하는 것입니다.

예를 들면, https://domain.co.kr이 아닌, https://domain.co.kr:444 등으로 특정 포트를 직접 지정하여 접속을 해야 하는 불편함이 존재합니다. 즉, 기존에는 IP 고갈이나 다른 포트로 접속 등으로 인해서 웹호스팅 서버 또는 한 대의 서버에서 여러 개의 도메인을 사용하는 서버에서는 https를 사용하는데 불편함이 있었습니다.

나. 멀티도메인(Multi-domain) SSL 인증서란?

멀티도메인 SSL 인증서는 기존의 단일 도메인 인증으로 인한 불편함을 없애고, 웹호스팅 서버나 한대의 서버에서 여러 개의 도메인을 운영하는 서버들에게 IP의 추가나 443이 아닌 다른 포트를 사용하지 않고 웹 서버 암호화 통신을 위해서 동일한 443 포트를 공유해서 사용할 수 있도록 편리함을 제공할 수 있는 인증서입니다.

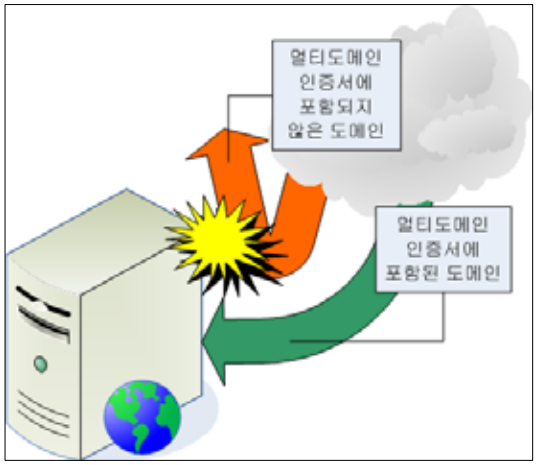
멀티도메인 SSL 인증서는 COMODO(http://comodo.com)라는 SSL 인증기관에서 제공하는 인증서입니다.

이 멀티도메인 SSL 인증서는 1년 동안의 인증기간을 갖는데, 그 인증기간 안에는 언제든지 최대 100개까지의 도메인을 추가할 수 있고, 또한 제거할 수도 있습니다. 기본적으로 도메인을 추가할 때는 도메인 하나당 추가 비용이 따로 청구되고 삭제 때는 청구가 되지 않지만, 실수로 삭제한 경우 다시 추가를 하려면 다시 비용을 내야 하므로 삭제시에는 주의해야 합니다.

2. 멀티도메인 SSL 인증서 구동 방식

멀티도메인 SSL 인증서가 여러 개의 도메인을 인식할 수 있는 방법은 CN(Common Name) 변수를 도메인 수만큼 여러 개 생성하여 인증서에 포함시켜 두고, 클라이언트가 암호화된 통신으로 웹 서버에 접속했을 때 웹 서버는 암호화 통신된 내용을 복호화하여, 클라이언트에서 요청한 도메인이 자신의 인증서에 있는 CN과 일치하는지 확인한 뒤 일치하면 보안 통신을 허가하는 방식으로 구동을 하게 됩니다.

이런 방법은 단일 도메인 인증서에서도 마찬가지로, 멀티도메인 인증서는 단지 인증서안에 여러 개의 CN을 포함하고 있다는 것 외에는 방식이나 기술의 차이는 없습니다.



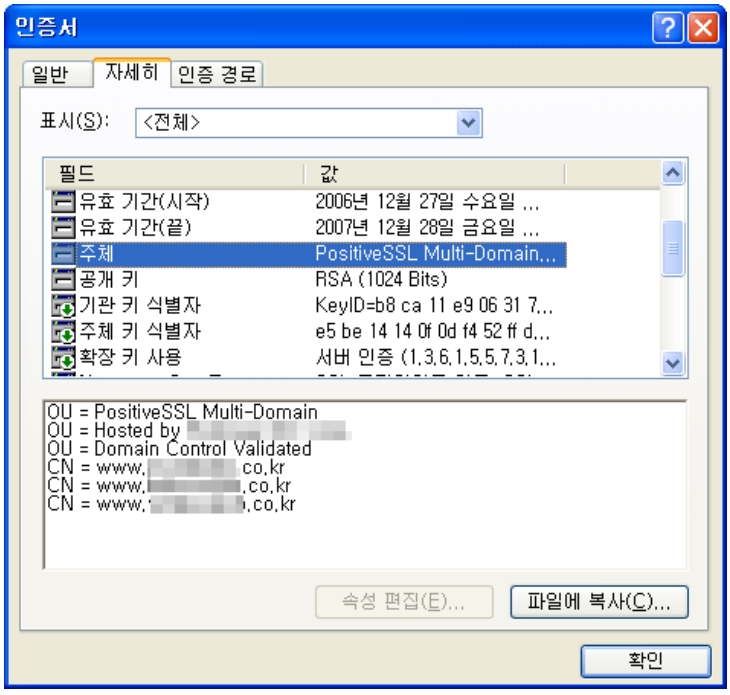
<그림 A-1> 멀티도메인 SSL 인증서의 CN이 있는 도메인과 없는 도메인의 동작

<그림 A-2>는 멀티도메인 SSL로 보호된 웹사이트 방문시에도 단일 도메인 SSL 방식으로 보호된 웹사이트와 마찬가지로 자물쇠 모양으로 현재 보안통신이 이루어지고 있음을 보여주고 있습니다.

<그림 A-2> 암호화 통신이 이루어지고 있음을 보여주는 자물쇠 이미지

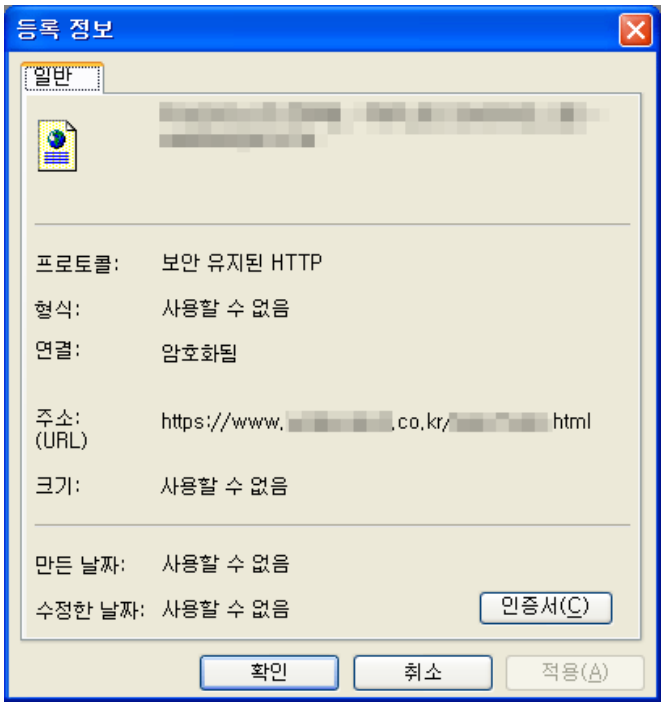
자물쇠 모양을 더블클릭하여 인증서를 자세히 보도록 하겠습니다.

인증서의 속성을 확인해보면 다수의 CN이 존재하는 것을 확인할 수 있습니다. 이 CN 변수에 값으로 설정된 도메인에 대해서만 암호화 통신이 가능합니다.



<그림 A-3> 다수의 CN이 포함된 멀티도메인 SSL 인증서

이미 단일 SSL이 적용된 웹페이지 속성에서 확인하였듯이 웹페이지의 속성을 확인하면 보안이 유지된 상태로 통신이 이루어지고 있음을 확인할 수 있습니다.



<그림 A-4> 보안이 적용된 웹페이지 속성 확인

3. 멀티도메인 SSL 인증서 적용 방법

가. 아파치 웹 서버

멀티도메인 SSL 인증서를 적용하기 위해서는 기존에 이미 설정되어 있던 평문 통신을 위한 http(80)의 가상호스팅 설정을 변경해야 합니다.

만일 평문 통신의 가상호스팅 설정을 변경하지 않으면, 보안통신을 위해서 https를 호출했지만 https가 응답을 하지 않고 일반 평문을 위한 http 프로토콜이 응답을 하는 등 에러가 발생할 수 있기 때문에, 평문은 http(80)에서, 암호화된 통신은 https(443)에서 호출하라고 확실히 구분을 해주어야 합니다. 아래의 <그림 A-5>과 <그림 A-6>이 아파치 서버의 설정파일(httpd.conf)에서 각각 평문 통신과 암호화 통신으로 구분한 설정 예입니다.

```
NameVirtualHost [redacted]:80

<VirtualHost [redacted].com:80>
        ServerAdmin webmaster@[redacted].com
        DocumentRoot /home/[redacted]/public_html
        ServerName [redacted].com
        ScriptAlias /cgi-bin/ "/home/[redacted]/cgi-bin/"
        ServerAlias [redacted].com www.[redacted].com
        ErrorLog /var/log/[redacted].com-error_log
        CustomLog /var/log/[redacted].com-access_log common
</VirtualHost>

<VirtualHost [redacted].co.kr:80>
        ServerAdmin webmaster@[redacted]com
        DocumentRoot /home/[redacted]/public_html
        ServerName [redacted].co.kr
        ScriptAlias /cgi-bin/ "/home/[redacted]/cgi-bin/"
        ServerAlias [redacted].co.kr www.[redacted].co.kr
        ErrorLog /var/log/[redacted].co.kr-error_log
        CustomLog /var/log/[redacted].co.kr-access_log common
</VirtualHost>
```

<그림 A-5> 아파치 서버에서 평문 통신을 위한 가상호스팅 설정

```
NameVirtualHost [redacted]:443

<VirtualHost www.[redacted].co.kr:443>
DocumentRoot "/home/[redacted]/public_html"
ServerName www.[redacted].co.kr
ServerAlias www.[redacted].com
ScriptAlias /cgi-bin/ "/home/[redacted]/cgi-bin/"
ServerAdmin webmaster@[redacted]
ErrorLog /usr/local/apache/logs/error_log
TransferLog /usr/local/apache/logs/access_log

SSLEngine on
SSLCipherSuite [redacted]
SSLCertificateFile /usr/local/apache/conf/ssl.crt/[redacted]
SSLCertificateKeyFile /usr/local/apache/conf/ssl.key/[redacted].key

<Files ~ "\.(cgi|shtml|phtml|php3?)$">
    SSLOptions +StdEnvVars
</Files>
<Directory "/usr/local/apache/cgi-bin">
    SSLOptions +StdEnvVars
</Directory>
SetEnvIf User-Agent ".*MSIE.*" \
         nokeepalive ssl-unclean-shutdown \
         downgrade-1.0 force-response-1.0
SetEnvIf User-Agent "LWP::" get_lost
SetEnvIf User-Agent "lwp-trivial" get_lost
CustomLog /usr/local/apache/logs/ssl_request_log \
          "%t %h %{SSL_PROTOCOL}x %{SSL_CIPHER}x \"%r\" %b"
</VirtualHost>
```

<그림 A-6> 아파치 서버에서 암호화 통신을 위한 가상호스팅 설정

나. IIS 웹 서버 (6.0 이상)

IIS에서 멀티도메인 SSL 인증서를 사용하는 방법은 아파치 서버보다 좀 더 까다롭습니다. IIS에서는 동일한 443 포트를 여러 도메인이 쓰고자 한다면, IIS의 80 포트에 대한 가상호스팅 설정을 하는 것이 아니라 SecureBindings라는 작업을 거쳐야 합니다.

SecureBindings는 443 포트를 여러 도메인이 쓸 수 있도록 설정해 주는 것을 뜻하며, 다음과 같은 절차를 통해서 설정합니다.

① 시작 → 실행 → 'cmd'(명령프롬프트) 명령 입력
명령이 정상적으로 수행이 되었다면, 그림과 같이 명령 프롬프트 창이 뜹니다.

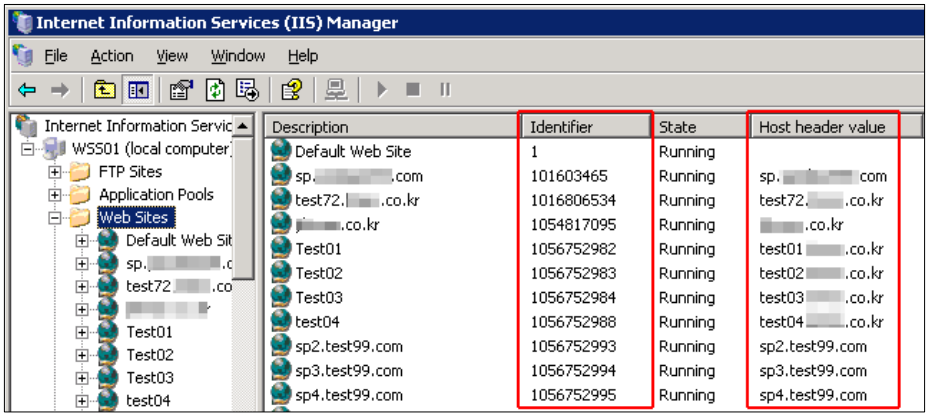
<그림 A-7> CMD command 실행 모습

② 다음 명령창에 SSL 호스트 헤더에 대한 SecureBindings의 메타베이스 속성을 설정하기 위해서 아래와 같은 형식의 명령을 입력합니다.

```
cscript.exe adsutil.vbs set /w3svc/<site identifier>/SecureBindings ":443:<host header>"
```

* <site identifier>는 도메인의 식별자 번호
* <host header>는 웹사이트의 호스트 헤더
* <그림 A-8> 참고

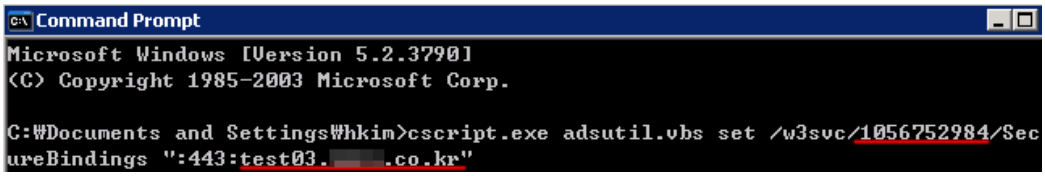
③ <그림 A-8>의 화면을 보기 위해서는 "시작 → 프로그램 → 관리도구 → Internet Information Service (IIS) 관리자"를 수행



<그림 A-8> IIS 관리자에서 Site Identifier와 Host header 값 확인

그림에서 첫 번째 빨간색 네모상자가 <Site Identifier>이고 두 번째 빨간색 네모상자가 <host header>입니다. <Site Identifier>는 시스템에 의해서 자동으로 부여되며, <host header>는 SSL 인증서를 신청할 때 사용하는 도메인 주소입니다.

④ 아래 <그림 A-9>는 <그림 A-8>의 test03.xxx.co.kr이 443을 사용하기 위해서 SecureBindings 메타베이스를 속성을 설정하는 과정입니다.



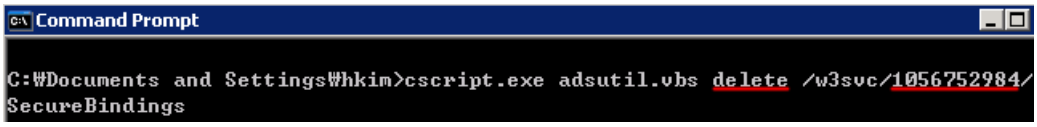
<그림 A-9> SecureBindings 메타베이스 추가

<그림 A-9>를 정상적으로 수행하면, test03.xxx.co.kr이 443 포트에 binding 됩니다. 그 결과를 <그림 A-10>에서처럼 IIS 관리자에서 확인해보면, test04.xxx.co.kr과 test03.xxx.co.kr이 암호화 통신을 위해서 같은 443 포트를 사용하고 있음을 확인할 수 있습니다.

Description	Identifier	State	Host header value	IP address	Port	SSL Por
Default Web Site	1	Running		* All Unassigned *	80	
sp.[redacted].com	101603465	Running	sp.j[redacted].com	* All Unassigned *	80	
test72.[redacted].co.kr	1016806534	Running	test72.[redacted].co.kr	* All Unassigned *	80	
[redacted].co.kr	1054817095	Running	[redacted].co.kr	* All Unassigned *	80	
Test01	1056752982	Running	test01.[redacted].co.kr	* All Unassigned *	80	
Test02	1056752983	Running	test02.[redacted].co.kr	* All Unassigned *	80	
Test03	1056752984	Running	test03.[redacted].co.kr	* All Unassigned *	80	443
test04	1056752988	Running	test04.[redacted].co.kr	* All Unassigned *	80	443

<그림 A-10> SecureBindings을 통한 443 포트 공유

만일 binding된 도메인을 지워야 할 경우에는 <그림 A-11>와 같이 delete 옵션을 이용하여 SecureBindings 메타베이스 속성에서 특정 사이트에 binding된 것을 해제(삭제)할 수 있습니다.



<그림 A-11> SecureBindings 제거

다. IIS 5.0에서의 멀티도메인 SSL 인증서 제약 사항

IIS 5.0에서는 SecureBindings을 지원하고 있지 않기 때문에, 멀티도메인 SSL 인증서를 같은 443 포트에서 사용할 수가 없습니다.

다만, IIS 5.0은 기존의 단독 SSL 방법과 같이 포트를 다르게 하는 방법을 사용해서 멀티도메인 SSL 인증서를 사용할 수 있습니다.

4. 멀티도메인 SSL 인증서의 문제점

지금까지 멀티도메인 SSL 인증서의 특징을 소개했는데, 다른 인증서들이 장단점을 가진 것과 마찬가지로 멀티도메인 SSL 인증서도 구조상 문제점을 가지고 있습니다.

① 모든 도메인 리스트 출력

같은 멀티도메인 SSL 인증서를 쓰는, 다시 말하면 한 대의 서버 내에서 운영되고 있는 https를 사용하는 모든 도메인이 노출될 수 있습니다.

② 동일한 만료기간

멀티도메인 SSL 인증서의 가장 큰 구조적 문제점이라고 볼 수 있는 것으로, 인증서 내에 포함되어 있는 각 도메인들의 SSL 서비스 만료 기간이 메인 인증서의 발급시기에 좌우된다는 것입니다.

다시 말하면, 멀티도메인 SSL 인증서의 최초 발급일자가 2007년 1월 1일 이라면, 1년이 되는 2008년 1월 1일에는 인증서에 속해있는 모든 도메인이 만료가 되어지는 것입니다.

예를 들어 최초 발급일자가 2007년 1월 1일인 멀티도메인 SSL 인증서에 2007년 1월에 포함된 도메인과 2007년 11월에 포함된 도메인이

모두 만료 일자가 2008년 1월 1일이라는 것입니다.

도메인마다 단독 SSL 인증서를 사용하면 각각 1년의 서비스 기간이 보증되는데 반해서(계약에 따라서 더 길수도 있다), 멀티도메인 SSL 인증서의 경우에는 일정 비용을 지불하지만, 1년이라는 서비스 기간을 보장받지 못할 수 있습니다.

③ 인증서 비용 문제

앞서 설명했듯이 도메인을 추가할 때마다 일정 비용을 지불해야 하고, 만료기간이 되어 멀티도메인 SSL 인증서를 갱신하고자 할 때에는 현재 멀티도메인 SSL 인증서에 포함되어 있는 모든 도메인에 대해서 각각 일정 비용을 지불해야 하므로, 늦게(만료일에 가깝게) 추가한 도메인에 대해서는 인증유지를 위해서 짧은 기간에 두 번의 비용을 지불해야 합니다.

5. 정리

이상으로 멀티도메인 SSL 인증서에 대해서 알아보았습니다.

멀티도메인 SSL 인증서는 인증서가 필요한 다수의 도메인을 한 대의 서버에서 운영해야 하는 경우 유용한 인증서 서비스입니다.

앞서 소개한 문제점을 충분히 고려하고 자신의 서비스 환경에 맞게 적절히 운영하면, 인증서가 필요한 다수의 도메인을 운영하고 있는 웹호스팅 서비스 제공 서버나 여러 도메인을 운영하는 서버의 여러 문제를 해결할 수 있는 방법이 될 수 있다고 생각됩니다.

부록 B. SSL 가속기 소개

1. SSL 가속기 정의

웹 브라우저 회사인 넷스케이프가 만든 SSL(Secure Socket Layer)은 널리 알려진 바와 같이 암호화와 복호화를 통해 데이터를 전달하여 안전한 통신을 가능하게 하는 기술입니다. SSL Key 교환과 Bulk 암호화 처리는 연산중심의 작업이기 때문에 웹 서버에서 SSL을 처리하는 전통적인 방식을 사용하면 서버의 CPU에 많은 작업 부하를 주게 됩니다. SSL 가속기 (SSL Accelator)는 웹 서버에 부하를 주지 않고 SSL 처리를 하기 위해 SSL 처리에 특화된 전용장비를 통해 처리하도록 하는 솔루션입니다.

2. SSL 가속기의 역할

대형 포털사이트와 같이 트래픽이 많으면서 SSL 방식의 보안서버를 운영하는 사이트가 SSL을 웹 서버에서 직접 운영하게 되면, 서버의 부하가 커지게 되고 사용자의 응답시간이 매우 느려질 수 있습니다. SSL 가속기는 SSL 연결에 따른 모든 암호화를 서버로부터 이관받아 독립된 장치에서 전담함으로써 서버의 부하를 감소시키고 웹 어플리케이션 시스템의 전체 성능을 향상시킬 수 있습니다.

또한 SSL 보안에 사용되는 서버의 암호화키를 SSL 가속기에 별도로 저장함으로써 암호화 키 유출에 따른 위험성을 줄일 수 있습니다. 서버 키가 유출될 경우 SSL 보안 통신 데이터가 해킹될 수 있고 허위 사이트나 피싱 사이트에 의한 피해가 발생할 수 있습니다.

그리고 SSL 인증서를 구입하여 웹 서버 대신 웹 가속기에 설치가 가능하기 때문에 기존에 운영중인 웹 프로그램을 수정할 필요가 없다는 장점을 가지고 있습니다.

3. SSL 가속기의 종류

① 1세대 SSL 가속기 'PCI / SCSI 카드타입'

SSL 가속기의 첫 번째 세대는 서비스를 제공하는 웹 서버가 작동하는 하드웨어에 직접적으로 설치가 되는 PCI나 SCSI 타입의 카드 형태의 제품들입니다.

PCI나 SCSI 카드 타입의 SSL 가속기 제품군은 우선 SSL 핸드쉐이크 과정을 담당함으로써 CPU에 부과되던 높은 부하를 절감시켰으며 설치 구조상 실제 콘텐츠 서비스를 수행하는 웹 서버 또는 어플리케이션 서버가 작동하는 하드웨어의 슬롯에 직접 장착되기 때문에 SSL 사용시 보호되어야 할 클라이언트 브라우저로부터 웹 서버 본체까지의 엔드 투 엔드(end-to-end)의 완벽한 보안이 이루어집니다. 그러나 이러한 제품군에는 치명적인 단점이 존재하는데 하나는 가속기 설치시 반드시 시스템의 중단이 필요하다는 것이고, 또 하나는 확장성의 문제입니다. 카드 타입의 가속기는 하나의 가속기가 하나의 서버만을 감당하는 물리적인 구성의 한계 때문에 여러 가지 이유로 인해 서버의 증설이 요구되는 경우 서버 증설 숫자만큼 가속기의 추가 구매 또한 필요하기 때문입니다.

② 2세대 SSL 가속기 'SSL 오프로더'

1세대 제품의 문제점을 개선하고 나온 2세대 SSL 가속기는 네트워크 장비 타입으로 흔히 SSL 오프로더(Offloader)라고 불리는 제품군입니다. SSL 오프로더는 기존의 카드타입 가속기들과 달리 하나의 가속기가 여러 대의 웹 서버나 애플리케이션 서버를 위한 SSL 가속기능을 수행함으로 기존 1세대 제품군의 확장성 문제를 보완했습니다. 웹 서버나 애플리케이션 서버와 분리된 설치 방법으로 인해 가속기 드라이버와 서버 하드웨어 충돌 등으로 인해 발생할 수 있는 문제의 소지를 없앴습니다.

그러나 네트워크 구성상 가속기와 서버사이에 물리적인 공백 구간이 있을 수 밖에 없고 브라우저가 발생시킨 암호화 패킷은 가속기에 복호화되고 클리어 텍스트(Clear Text)로 이 구간을 통과해 서버에 전달되기 때문에 실제 클라이언트 브라우저로부터 서버까지의 엔드 투 엔드 보안이 불가능합니다. 또한 일반적인 인라인 구성(In-line Configuration)의 경우 설치시 서비스의 일시적 중단을 피할 수는 없습니다.

몇 가지 단점에도 불구하고 최근에는 단독 장비 제품형태의 2세대 제품군이 주류를 이루어 시장을 선도하고 있습니다. 또한 2세대 제품군이 가지는 단점을 보완한 장비들도 개발되고 있습니다. 특히 백엔드(Back end) SSL 기능을 통한 엔드 투 엔드 보안 제공이나 원 암 구성(One-arm Configuration)을 통한 서비스 중단없는 설치기능 등을 통해 보다 효과적으로 SSL 가속기를 실제 네트워크상에 구현할 수 있는 다양한 방법들이 나오고 있습니다.

※ 참고문헌

1. SSL 프로토콜에 대한 이해, 퓨쳐시스템 http://www.future.co.kr/
2. 보안서버 가속을 위한 SSL 가속기 솔루션, 어레이네트웍스 코리아
3. SSL Accelerator for Secure Web Server, 엑스비전씨큐리티시스템

보안서버 구축 가이드 개발을 위하여 다음과 같은 분들께서 수고하셨습니다.

2007년 2월

총괄책임자	정보통신부 개인정보보호팀	팀 장 정현철
	한국정보보호진흥원 정책개발단	단 장 정경호
사업참여자	정보통신부 개인정보보호팀	사 무 관 서정훈
	한국정보보호진흥원 기술정책팀	팀 장 김성훈
	한국정보보호진흥원 기술정책팀	선임연구원 최광희
	한국정보보호진흥원 기술정책팀	선임연구원 정승욱
	한국정보보호진흥원 기술정책팀	연 구 원 이민우
검 토 (가나다순)	이니텍 기술기획팀	
	한국전자인증 인증서비스&기술지원팀	
	한비로 서버운영팀	
	호스트웨이 IT Consulting Team / Business Development Team	